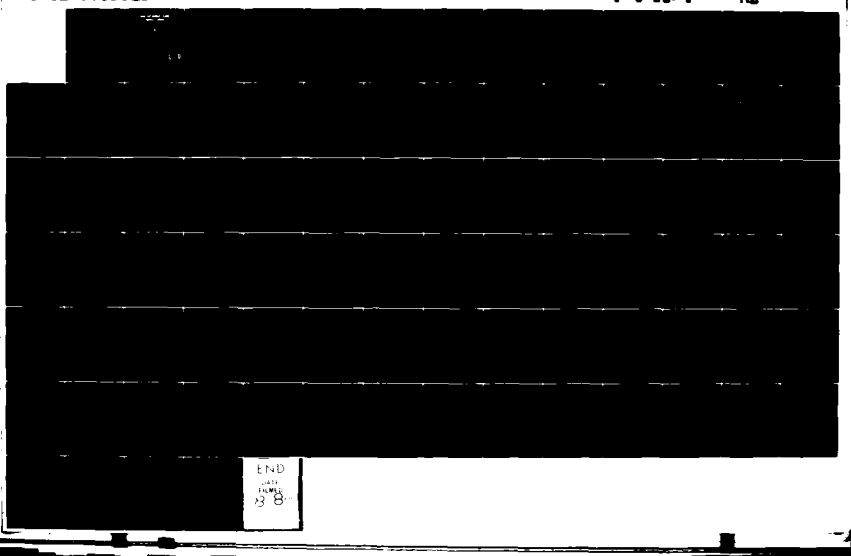
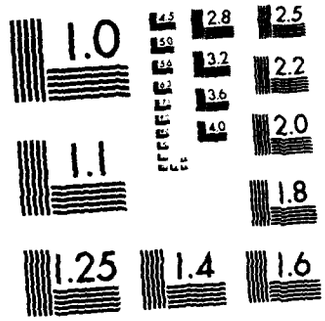
AD-A194 344 REFINEMENT AND EXTENSION OF SHRINKAGE TECHNIQUES IN LOSS RATE ESTIMATION O..(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA C R DICKINSON MAR 88 1/1

UNCLASSIFIED F/G 12/4 NL



END
DATE FILMED
3 88



NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

REFINEMENT AND EXTENSION OF SHRINKAGE TECHNIQUES IN LOSS RATE ESTIMATION OF MARINE CORPS OFFICER MANPOWER MODELS

by

Charles R. Dickinson

March 1988

Thesis Advisor: R. R. Read

REPORT DOCUMENTATION PAGE

1a Report Security Classification: Unclassified			1b Restrictive Markings
2a Security Classification Authority			3 Distribution Availability of Report: Approved for public release; distribution is unlimited.
2b Declassification Downgrading Schedule			
4 Performing Organization Report Number(s)			5 Monitoring Organization Report Number(s)
6a Name of Performing Organization: Naval Postgraduate School	6b Office Symbol (*if applicable*): 30		7a Name of Monitoring Organization: Naval Postgraduate School
6c Address (*city, state, and ZIP code*): Monterey, CA 93943-5000			7b Address (*city, state, and ZIP code*): Monterey, CA 93943-5000
8a Name of Funding Sponsoring Organization	8b Office Symbol (*if applicable*)		9 Procurement Instrument Identification Number
8c Address (*city, state, and ZIP code*)			10 Source of Funding Numbers: Program Element No / Project No / Task No / Work Unit Accession No

11 Title (*Include security classification*): REFINEMENT AND EXTENSION OF SHRINKAGE TECHNIQUES IN LOSS RATE ESTIMATION OF MARINE CORPS OFFICER MANPOWER MODELS

12 Personal Author(s): Charles R. Dickinson

13a Type of Report	13b Time Covered	14 Date of Report (*year, month, day*)	15 Page Count
Master's Thesis	From To	March 1988	87

16 Supplementary Notation: The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms (*continue on reverse if necessary and identify by block number*)
Field	Group	Subgroup	attrition estimation, Bayes, Miller transformation inverse.

19 Abstract (*continue on reverse if necessary and identify by block number*)

This thesis is a continuation of previous work to apply modern multiparameter estimation techniques to the problem of estimating attrition rates for a large number of small inventory cells in manpower planning models used by the U.S. Marine Corps. The main advances involve the promising introduction of empirical Bayes (non-constant shrinkage) techniques, recognition of the non symmetric nature of the errors with a response to this, and some insight into all aggregation plans that should help provide greater stability for the estimation methods. In addition, the roles of some middle level methodological choices are explored.

20 Distribution Availability of Abstract: ☒ unclassified unlimited ☐ same as report ☐ DTIC users	21 Abstract Security Classification: Unclassified	
22a Name of Responsible Individual: Robert R. Read	22b Telephone (*include Area code*): (408) 646-2382	22c Office Symbol: 55Re

DD FORM 1473,84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

Approved for public release; distribution is unlimited.

Refinement and Extension of Shrinkage Techniques in Loss Rate Estimation of Marine Corps Officer Manpower Models

by

Charles R. Dickinson
Captain, United States Marine Corps
B.S., United States Naval Academy, 1978

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1988

Author: _____
Charles R. Dickinson

Approved by: _____
Robert R. Read, Thesis Advisor

_____
Paul R. Milch, Second Reader

_____
Peter Purdue, Chairman,
Department of Operations Research

_____
James M. Fremgen,
Acting Dean of Information and Policy Sciences

ABSTRACT

This thesis is a continuation of previous work to apply modern multiparameter estimation techniques to the problem of estimating attrition rates for a large number of small inventory cells in manpower planning models used by the U.S. Marine Corps. The main advances involve the promising introduction of empirical Bayes (non-constant shrinkage) techniques, recognition of the non symmetric nature of the errors with a response to this, and some insight into all aggregation plans that should help provide greater stability for the estimation methods. In addition, the roles of some middle level methodological choices are explored.

(Keywords: thesis; Miller transformation inverse).

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

For my wife, Dorie, who basically functioned as a single parent during the preparation of this thesis. By freeing me to work as many hours as needed, you made this effort as painless as possible. Your contributions did not go unnoticed.

I. INTRODUCTION

A. PURPOSE

In support of project OPUS (Officer Planning and Utility System), Headquarters, U.S. Marine Corps, in 1985, requested assistance in exploring new methods of generating manpower loss rates in order to improve upon the currently used method of force rate analysis. In response, students and faculty in the Operations Research department at the Naval Postgraduate School began exploring the use of modern multiparameter estimation techniques for this problem. Special emphasis is placed on the *small cell* problem, i.e., categories of officer skill, grade and length of service which have low inventory figures. Historically, rate estimators for small cells are unstable and a large number of these cells exist.

This paper builds upon previous work on this problem. The main advances are to append new measures of effectiveness as requested by the Navy Personnel Research and Development Center (NPRDC), introduce a class of empirical Bayes estimators, and explore the effects of some middle level choices in applying the new and existing methods. The ultimate goal is to refine the techniques presented here in order to validate a clear policy for predicting loss rates.

B. BACKGROUND

For an introduction into Marine Corps policy concerning manpower planning, the reader is referred to a thesis submitted in September 1985, at the Naval Postgraduate School, by Major D.D. Tucker [Ref. 1], who presents a detailed background into the Marine Corps officer structure and the manpower planning process. Tucker also provides basic attrition rate theory and calculates overall attrition rates in several different formats as they pertain to the Marine Corps. As the aggregation rate begins to grow smaller with refinement, Tucker illustrates the irregular behavior of losses due to voluntary attrition and the small numbers of losses obtained in such aggregates. Tucker [Ref. 1:p. 50] introduces the James-Stein technique of loss rate estimation as a *shrinking* of individual cell averages toward a grand average in order to reduce the risk, or squared differences of forecast and actual values, the goal being an improvement over the classical maximum likelihood estimator, referred to from now on as the MLE. Tucker chose the ranks of First Lieutenant and Lieutenant Colonel for evaluation using military occupational specialty (MOS) groups of combat support and ground combat (three MOS's were selected for each group). These aggregation schemes were carried through other studies and are included for historical purposes in this paper.

Another thesis submitted at the Naval Postgraduate School in March 1986, by Major J.R. Robinson [Ref. 2] focused on a technique called limited translation of the James-Stein estimator. This particular model attempts to minimize the risk mentioned above by reducing the shrinkage of rate estimates for cells which fall outside a certain range of values centered on the grand mean. It was thought that shrinking all cells by the same rate toward this grand mean may be unwise since the attrition rates for those particular cells, those farthest from the grand mean, may contain attributes which are different than the majority of the cells in the aggregate. Additionally, Robinson's work carries forward the study done by Tucker concerning MLE and James-Stein estimators.

These feasibility studies were limited in scope due to the format of the summary date tape available (acquired from NPRDC), which allows only coarse cell definition. A cell is defined to be a cross-classification of the forty military occupational fields (OF), thirty-one lengths of service (LOS), and ten grades for a total of 12,400 categories for manpower planning purposes [Ref. 2:p. 10]. The data does not distinguish between limited duty officers, those officers specifically designated for limited duty within certain MOS's, and unrestricted officers (regular and reserve) [Ref. 1:p. 20]. These factors dictated a broad aggregation scheme that created stable cell inventories and

allowed the common variance assumption crucial to their work. But some of the application models require more refined information along those lines and experiments with finer aggregation levels, necessary in some real cases, caused this assumption to be violated.

Tucker and Robinson did conclude that the current scheme of attrition rate prediction could be improved upon by their methods, although no dominant scheme was uncovered. Tucker and Robinson did identify problem areas in their studies that required additional work. Tucker [Ref. 1:p. 71] cited a need for a better aggregation method to produce a more homogeneous attrition behavior in all cells. Also, small probabilities of loss within a cell were not dealt with successfully. Robinson [Ref. 2:p. 32] revealed the small cell problem and the inability to normalize the cell means or stabilize the variance with a data transformation. The recommendations of the previous studies highlighted the need for a more refined data tape with current information to include, among others, full Military Occupational Specialty (MOS) information, grade separation to include regular/reserve status, promotion zone data and breakdown of attritions by type.

C. PROGRESS

The project of loss rate estimation, sponsored by NPRDC, is currently moving into the implementation stage. An operational data tape is now available which includes

detailed information on Marine Corps officers for the years 1977-86. Work has begun at the Naval Postgraduate School to break out the data in a useful format and its availability in now nearing completion.

In addition to the results stated earlier, Tucker and Robinson provided loss rate estimates for small cells with no attrition, i.e., MLE equal to zero, and risk effects of the different schemes studied. These results provided the springboard for this study and the requirements for additional future studies.

Captain R.W. Larsen [Ref. 3] has developed a promising aggregation scheme based on cluster analysis, i.e., a classification scheme to aggregate cells which reflect a greater degree of homogeneity in attrition rates than are allowed using current aggregation methods. Table 1 displays the resulting scheme, given in year of current service (YCS) groups proposed for implementation by Larsen.

Table 1. AGGREGATION METHOD PROPOSED BY LARSEN

MOS Category	Bounded YCS Groups
Fixed-Wing Pilots	(1-6,8-19) (7) (20-25) (26)
Rotary-Wing Pilots	(1-5,8-19) (6,7) (20-25) (26)
Naval Flight Officers	(1-5,8-19) (6,7) (20-25) (26)
Lawyers	(1-6,8-19) (7) (20-25) (26)
All Else	(1-3,6-19) (4,5) (20-25) (26)

The parentheses encompass YCS groups which behave similarly. In each case the grade is fixed, i.e., one should aggregate over YCS before aggregating over grade.

This information will allow for more successful application of the loss rate estimation techniques of interest here when the new data tape is provided.

D. GOALS

In an attempt to refine the methods of attrition rate estimation presented in the previous pilot studies, the following goals were set for this paper:

1. Modify the existing methods to:
   a) use the refined data format,
   b) extend the work to the case of unequal or non constant variance in the small cell inventory problem,
   c) study alternate transformation inversion techniques,
   d) introduce and evaluate additional measures of effectiveness suggested by NPRDC to include, but not limited to, cross validation underages and overages of the deviation between forecasts and actuals,
   e) examine the graphical effect of shrinkage in the original scale.

2. Introduce empirical Bayes estimation method in several forms for attrition rates for consideration as another option to solve the problem.

II. EXTENSIONS TO ESTIMATION METHODS

A. GENERAL

As stated in Chapter I, Tucker and Robinson published pilot studies introducing the James-Stein, limited translation James-Stein, Maximum Likelihood and Transform Scale Cell Average estimators as alternatives to the aggregate methods previously used for manpower planning in the Marine Corps. Their performance was promising, and this chapter will investigate ways to enhance their performance and possibly develop a dominant estimator for future attrition prediction. Additionally, NPRDC has expressed interest in additional measures of effectiveness for these estimators which may lead to a sharper direction for the goal of producing a viable estimation policy. This chapter concentrates on the following issues concerning previous studies.

B. NON CONSTANT VARIANCE

For a cell having an inventory n, and attrition rate p, the number of attritions y is described by the Binomial (n,p) distribution. The variance of the estimator of $\hat{p} = y/n$ is given by the familiar formula

$$\text{var}(\hat{p}) = \frac{p(1-p)}{n} \qquad (2.1)$$

The Freeman-Tukey double arcsine transformation was used by Tucker [Ref. 1:p. 55] and Robinson [Ref. 2:pp. 74-75] in order to map the raw losses to the transformed scale. This formula is given by

$$x = 0.5[n + 0.5]^{1/2}\{\sin^{-1}[\frac{2y}{(n + 1)} - 1] + \sin^{-1}[\frac{2(y + 1)}{(n + 1)} - 1]\} \qquad (2.2)$$

Note that n=n(t) may change with time and thus, so would x. The question of averaging over time before or after applying the Freeman-Tukey transformation is not discussed in this paper. One of the goals of using this transformation is to stabilize the variance of this estimate. In the work of Tucker and Robinson, the assumption of normally distributed random variables with common variance was the setting in which the James-Stein estimator was expected to perform well. Indeed, for n moderate or large and p not too extreme, the variance of the Freeman-Tukey transformation is approximately one. But as discovered by Robinson, this assumption does not hold well when applied to cells with small inventory. A more careful look at the variance of the transformed data is taken here, with particular interest in the region of unstable variance where n and p are small.

Some exploratory graphical work has revealed that the natural log of the variance of x, the transformed value,

behaves quite well as a linear function of μ(=E[X]) and μ-1 in this unstable zone. This leads to the interpolatory formula:

$$\text{var}(x) = 1.6835\mu^{-.8934}(\mu - 1)^{.9881}, \qquad (2.3)$$

$$1.001 \leq \mu \leq 2.2$$

The upper limit, $\mu = 2.2$, is the value for which var(x) = 1. Since the least squares fitting process was applied to the logarithms, μ can not be allowed to fall to one or below. The lower limit, $\mu = 1.001$, is an arbitrary value that meets this constraint. The coefficients were validated by a linear regression on the natural log of this formula. The actual variance function for n = 1,2,3,4,5,7,10 appear in Figure 1, and the fit is remarkably good for $n \geq 3$. The upper graph shows the region of special interest for this paper and this portion of the curve is fitted by the formula in Equation 2.3 above. The lower plot exhibits the outer tale effect yet to be explored. The small values of n(<3) do not maintain the stable variance region, i.e., var[x]=1, for long before falling back. As the value for n increases, the more stable the variance becomes. Table 2 is a partial display of the residuals for selected n values, fifteen equally spaced values in the range $1.001 \leq \mu \leq 2.2$, computed by the linear regression formula above. For values of $n > 2$ the difference between actual and fitted values are considered acceptable for this work.

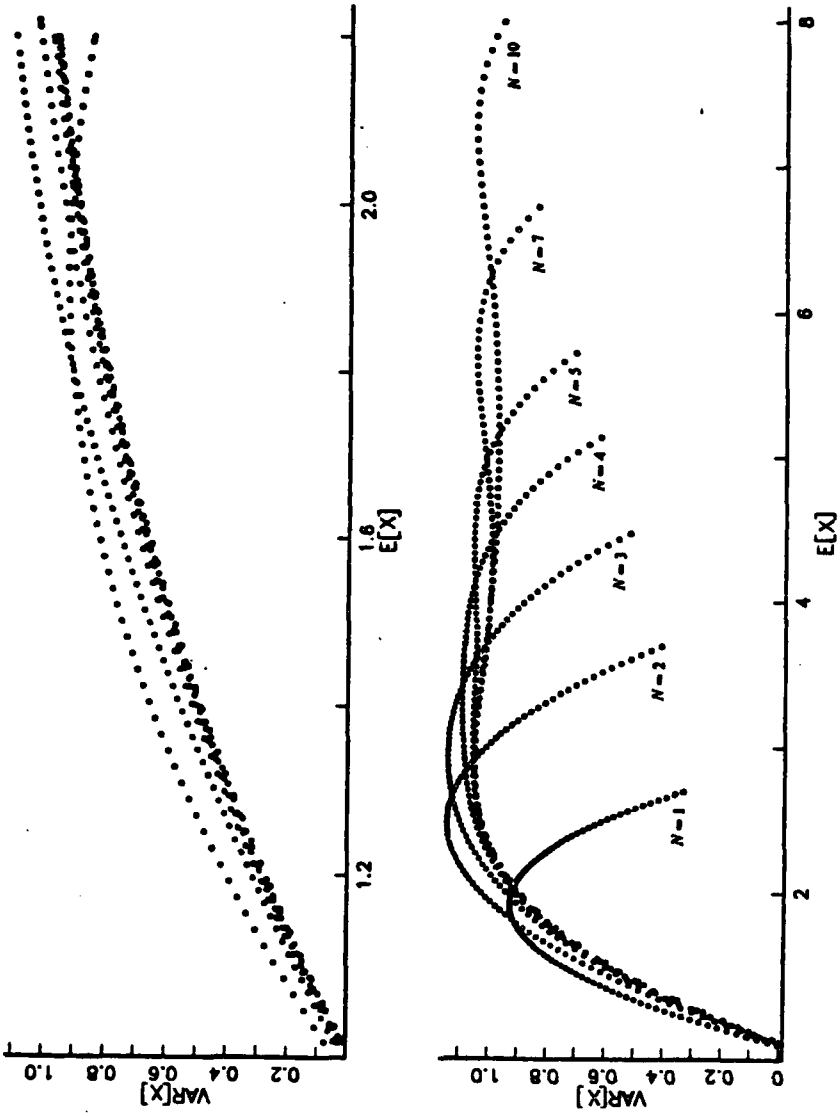


Figure 1. Variance of Freeman-Tukey Transform as Function of It's Mean for Selected n.

Table 2. RESIDUAL LISTING FOR SELECTED VALUES OF N

N=1	N=2	N=4	N=5	N=10
0.07191	0.03912	0.02139	0.017	0.008313
0.08466	0.02569	0.002853	-0.001695	-0.01106
0.1033	0.0239	-0.005448	-0.01049	-0.0206
0.122	0.03002	-0.00655	-0.01185	-0.02257
0.1385	0.04068	-0.004026	-0.009566	-0.02067
0.15	0.05388	0.0000676	-0.005727	-0.0166
0.1552	0.06853	0.004027	-0.002103	-0.01291
0.1527	0.08234	0.007447	0.0009541	-0.009981
0.1414	0.09481	0.00948	0.002557	-0.008299
0.1209	0.1055	0.009825	0.002448	-0.008128
0.08958	0.1129	0.008369	0.0004583	-0.009929
0.04832	0.1169	0.004997	-0.003511	-0.01327
-0.005254	0.1171	0.0001116	-0.00893	-0.0182
-0.06914	0.113	-0.006798	-0.01654	-0.02463
-0.1466	0.104	-0.01534	-0.02585	-0.03366

The boundaries for μ, in the unstable region, as a function of n and p are provided in Figure 2. The formula derived was quite adequate for this study.

C. ALTERNATE TRANSFORM INVERSION METHODS

There are three issues in the inversion of the data from the transformed space. The first is that of selecting values of n, i.e., an average of the n(t)'s appearing in the estimation year inventories. This issue is discussed later in the chapter, with the development of a simulation for comparing mean inventory values. The second is that of the inversion of the sum of the two arcsine functions. The third involves the case of time averaging in the transformed scale and the question of what matching form of average

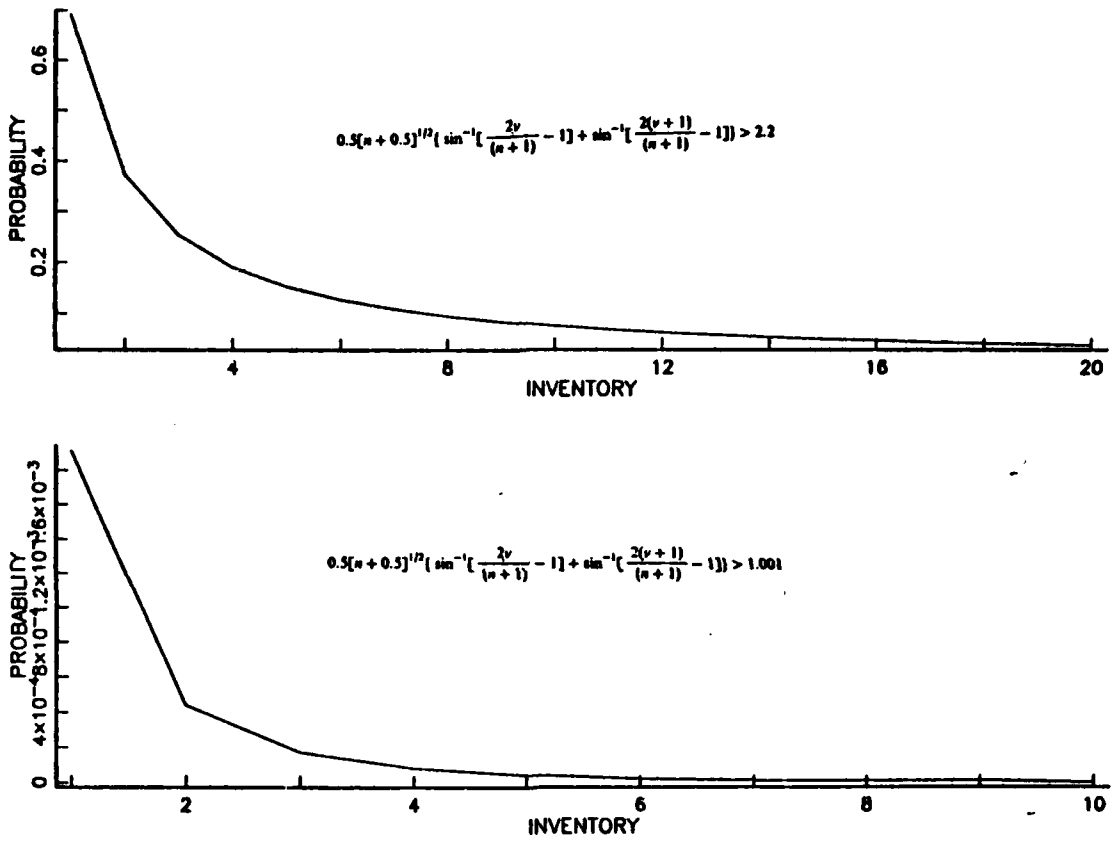


Figure 2. Envelope for the Non-Constant Portion of the Fitted Variance as Functions of Attrition Rate and Inventory.

inventory should be used in the inversion. The third issue is not covered in this paper, and is still pending. Treatment of the second issue will be provided first.

The method used in the pilot studies of Tucker and Robinson, and the most transparent transform inversion formula is:

$$\hat{p}_i = 0.5[1 - \sin(\frac{x}{(n + 1/2)^{1/2}})] \qquad (2.4)$$

This will be referred to as the Basic Inversion. The formula offered by Carter and Rolph [Ref. 4] is given by:

$$\hat{p}_i = \sin^2(\hat{\theta}_i) + [\frac{(1 - 2\sin^2\hat{\theta}_i)}{(4(n/k) + 2)}]\hat{B}_i \qquad (2.5)$$

where $\hat{\theta}_i$ is $x/(n + 1/2)^{1/2}$ and $\hat{B}_i$ is the estimated amount of shrinkage in the transformed scale depending on the Bayes model used (constant or proportional prior), k is the number of cells and n is inventory.

Apparently, Carter and Rolph invert prior to shrinking and the above formula provides a shrinkage in the original scale. Rewriting the above, it appears to shrink towards $\hat{p} = 1/2$, and is given by:

$$\hat{p}_i = 1/2(\frac{2\hat{B}_i}{4(n/k) + 2}) + \sin^2(\hat{\theta}_i)(1 - \frac{2\hat{B}_i}{4(n/k) + 2}) \qquad (2.6)$$

An exact Freeman-Tukey inversion formula developed by Miller [Ref. 5] was recently uncovered. It is exact in the sense that for a single cell, the formula given by:

$$\hat{p}(t) = 0.5\{1 - \mathrm{sgn}(\cos t)[1 - (\sin t + \frac{\sin 5 - 1/\sin t}{n})^2]^{1/2}\} \qquad (2.7)$$

where $t = x/(n + 1/2)^{1/2} + \pi/2$ and a is the average central inventory, returns the empirical rate y/n. This will be referred to as the Freeman-Tukey Exact (FTE) inversion. Since this formula is applied using an average value of n, there may be concern that it oscillates between successive integer values of n. Figure 3 indicates, using values of n=5 and 10, that it is quite smooth as an interpolation formula. In the article, Miller suggested using the harmonic mean of the n values provided by the time changes. This issue is examined later in this chapter. The Miller formula was incorporated in the Marine Corps Inventory Attrition Analysis Program provided in Appendix B.

D. SHRINKAGE VISUALIZATION

When shrinkage is a fixed amount for all cells, it is readily visualized, for all degrees of shrinkage, by the linear diagrams presented in the works of Tucker [Ref. 1:p. 52] and Casella [Ref. 6]. Robinson [Ref. 2:p. 20] also presented diagrams of the limited translation option and how the shrinkage is affected. All of these diagrams refer to the transformed scale.

Of more immediate interest is the question of how to visualize shrinkage diagrams of this type in the original scale. Illustrations of diagrams of this type are provided

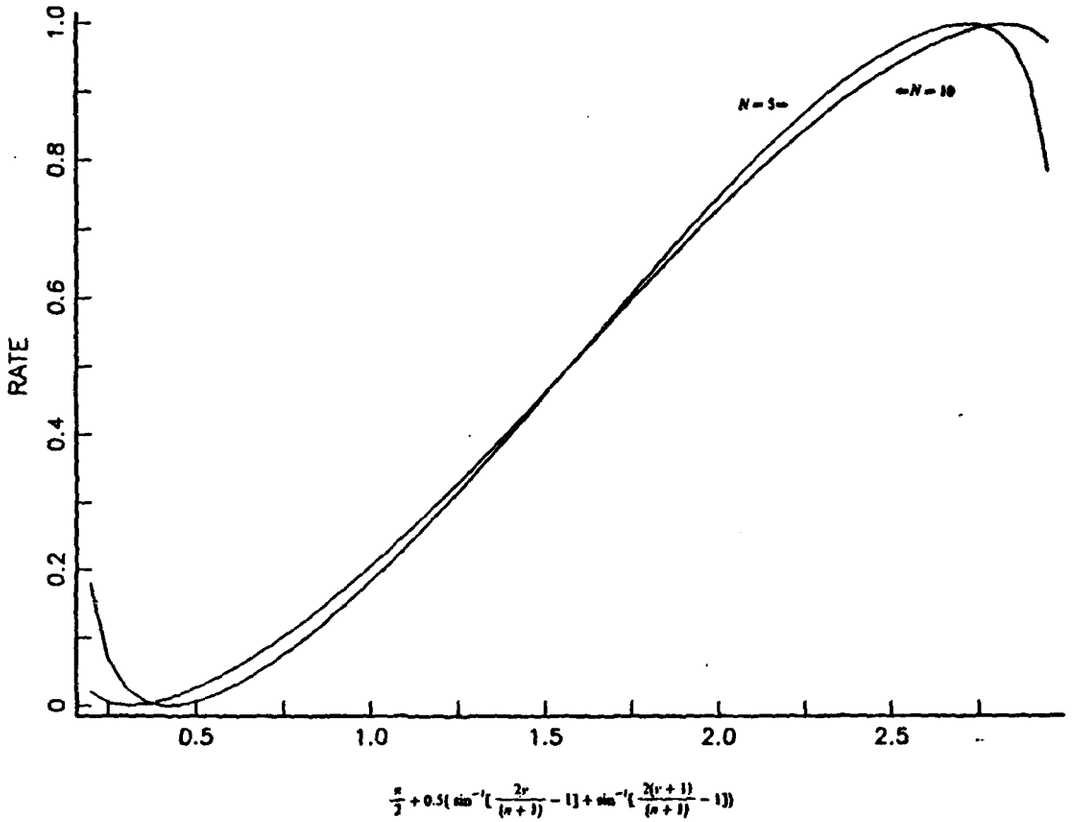


Figure 3. Suitability of the Exact Freeman-Tukey Inversion as an Interpolatory Formula.

using both the basic inversion formula used by both Tucker and Robinson, and the FTE formula. Figure 4 shows how the diagram by Casella [Ref. 6:p. 84] looks in the original scale using the same value of n for all cells. An interesting note to consider is the effect of the difference between inversion formulas, i.e., in the graph which combines the Basic and FTE inverses, the FTE inversion tends to shrink at a slower rate, as values get further from the mean. For a modest transition into the case of non constant n, Figure 5 provides for two values of n. Finally, Figure 6 illustrates the point with many different values of n, taken from actual Marine Corps officer inventory data. In this case, note that Basic inversion shrinks to one original scale value while the FTE inverse is not as focused.

E. CHOICE OF AVERAGE INVENTORY VALUES OVER TIME

A Fortran simulation was developed, included as Appendix C, to test the performance of each of several common means (arithmetic, geometric and harmonic) in the comparison of the inverted estimates with the empirical y/n. Values of n were produced from a Poisson distribution with rate λ ($1 \leq \lambda \leq 20$), and attrition values, y, were produced from the Binomial (n,p) distribution where a was in the range $.01 \leq p \geq .40$. The n and y values were then transformed by using the Freeman-Tukey formula. An exhaustive study of inversion variations with both the FTE formula and the basic formula

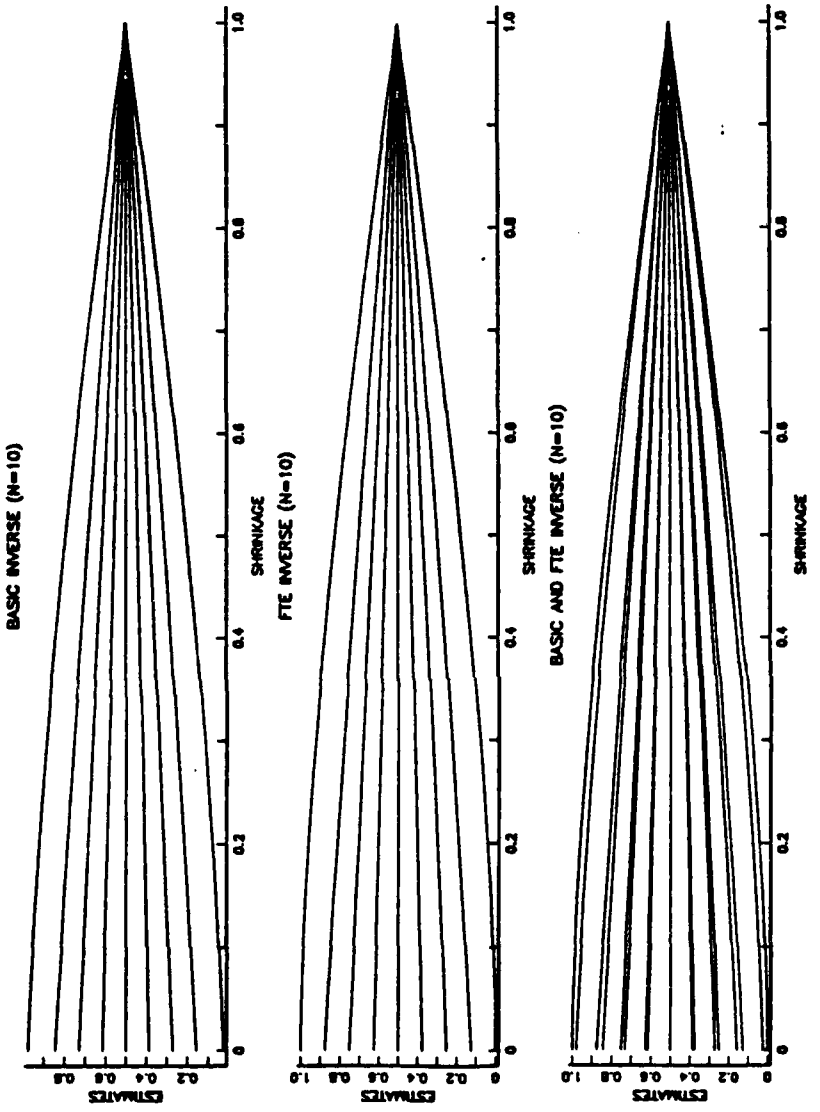


Figure 4. Non Linear Shrinkage Diagrams for a Single Value of Inventory.

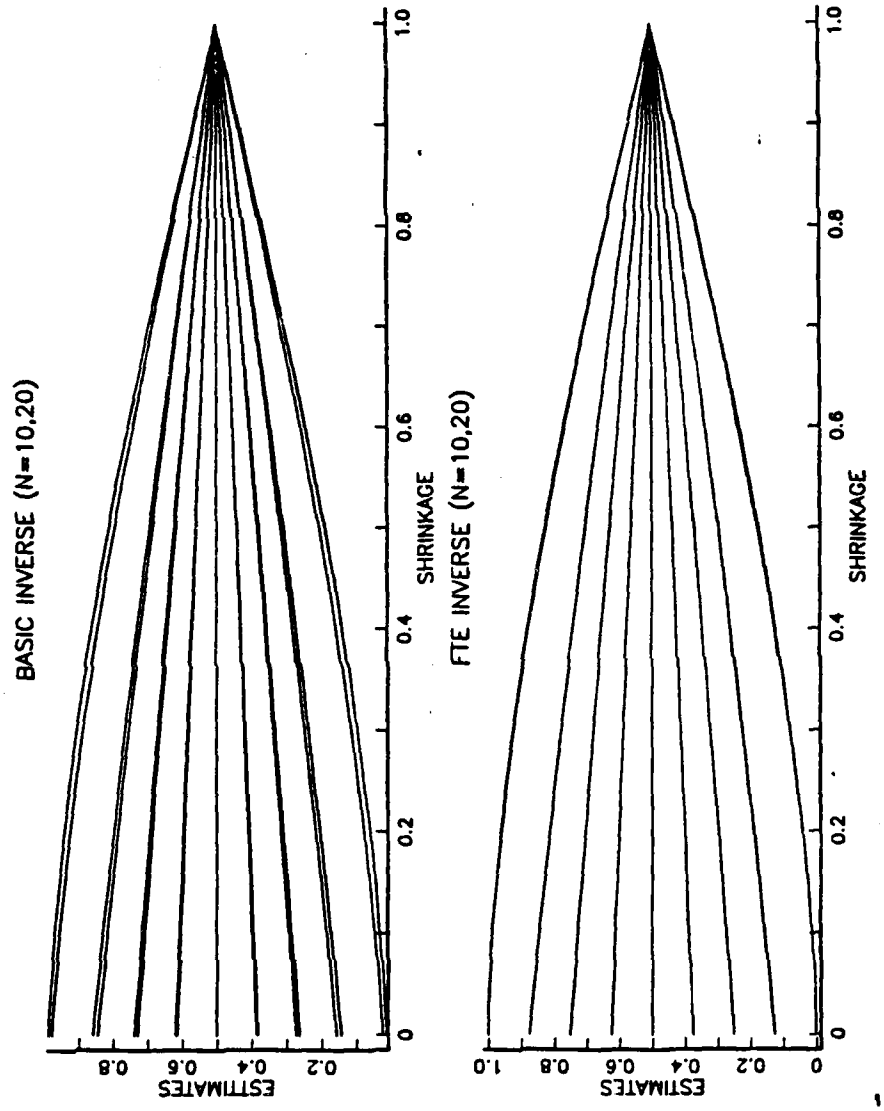


Figure 5. Non Linear Shrinkage Diagrams for Two Values of Inventory.

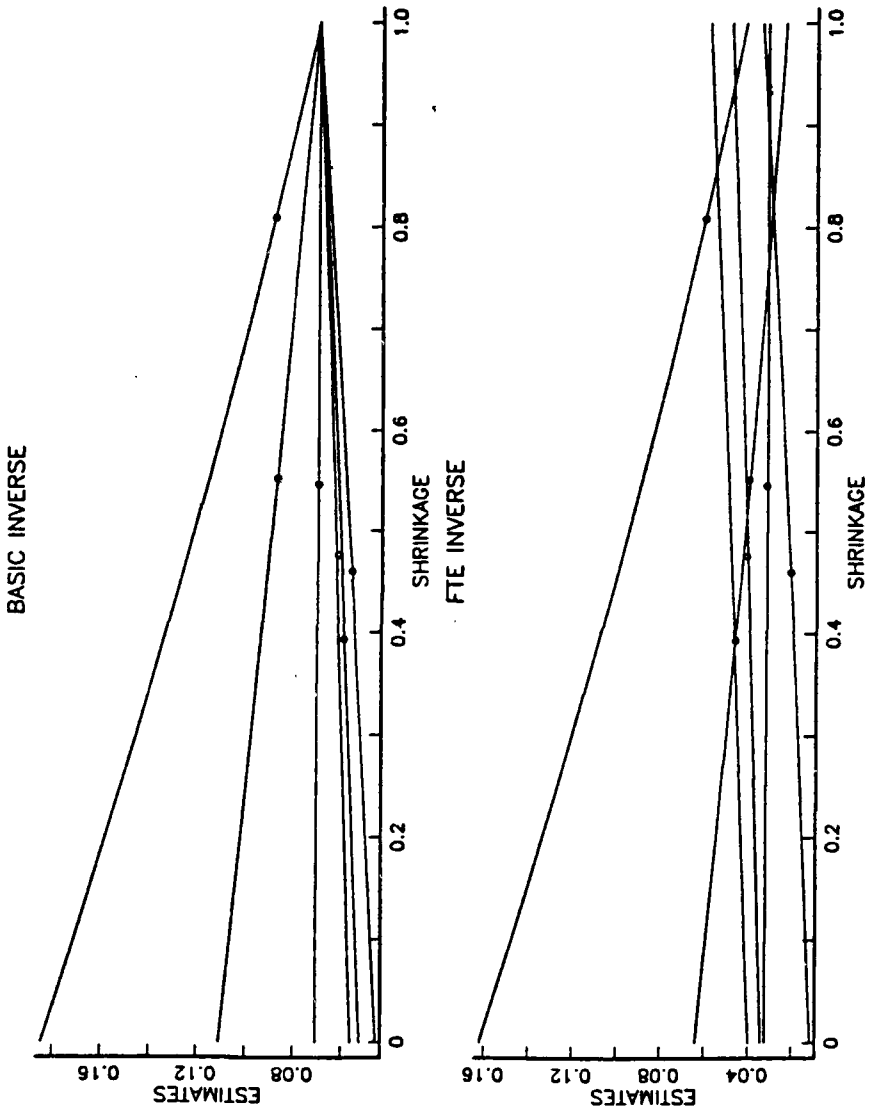


Figure 6. Non Linear Shrinkage Diagrams with Empirical Bayes Estimators for Test Case.

used by Robinson were compared. Table 3 contains selected outputs which are typical of the broader study and shows how the means performed against each other in three different computations listed below.

1. The BASIC Method utilizes the Basic inversion formula and the average transformed scale value.

2. The FTE1 Method utilizes Equation 2.7 with the t values equal to the average (over replications, i.e., time) of $x/(n + 1/2)^{1/2}$ and x given by Equation 2.2. More explicitly, see Equation A.16.

3. The FTE2 Method utilize Equation 2.7 with the t values equal to the average (over replication, i.e., time) of x directly, and x given by Equation 2.2 followed with a division by $(n + 1/2)^{1/2}$

In the context of this study, the arithmetic mean tended to perform as well or better than the geometric or harmonic means. The choice of arithmetic mean also circumvents the problem of an undefined term when $n_i = 0$ is encountered while using the harmonic of geometric means.

F. QUESTIONS RELATING TO VALIDATION

The pilot studies by Tucker and Robinson both used four years (1977-80) for estimation of the attrition rates and the remaining three years (1981-83) for validation. The estimation rates were used in an attempt to forecast the values for the three validation years. The quality of the forecasts deteriorates as lead time increases in the future. This was anticipated, in fact, Rowe et al., at NPRDC have shown a time series effect in their work forecasting attritions in the Navy [Ref. 7] and are currently

Table 3. SIMULATION OUTPUT OF MEAN VALUES FOR INVERSION UTILIZATION

λ	REPS	P	METHOD	ARITHMETIC	GEOMETRIC	HARMONIC
2.0	40	0.05	BASIC	0.1068	0.1176	0.1308
			FTE1	0.0252	0.0189	0.0122
			FTE2	0.0181	0.0208	0.0245
2.0	40	0.10	BASIC	0.1399	0.1537	0.1708
			FTE1	0.0535	0.0458	0.0369
			FTE2	0.0489	0.0558	0.0648
2.0	40	0.20	BASIC	0.2402	0.2629	0.2906
			FTE1	0.1900	0.1829	0.1739
			FTE2	0.1657	0.1869	0.2144
2.0	40	0.40	BASIC	0.3713	0.4038	0.4429
			FTE1	0.3442	0.3403	0.3353
			FTE2	0.3325	0.3715	0.4213
4.0	40	0.05	BASIC	0.0750	0.0815	0.0921
			FTE1	0.0235	0.0200	0.0147
			FTE2	0.0200	0.0221	0.0258
4.0	40	0.10	BASIC	0.1280	0.1387	0.1564
			FTE1	0.0768	0.0724	0.0655
			FTE2	0.0726	0.0798	0.0922
4.0	40	0.20	BASIC	0.1965	0.2125	0.2386
			FTE1	0.1446	0.1406	0.1339
			FTE2	0.1485	0.1628	0.1869
4.0	40	0.40	BASIC	0.4382	0.4694	0.5185
			FTE1	0.4564	0.4558	0.4549
			FTE2	0.4280	0.4638	0.5223
8.0	40	0.05	BASIC	0.0552	0.0580	0.0616
			FTE1	0.0181	0.0166	0.0148
			FTE2	0.0156	0.0165	0.0177
8.0	40	0.10	BASIC	0.1226	0.1285	0.1363
			FTE1	0.0922	0.0904	0.0880
			FTE2	0.0828	0.0873	0.0933
8.0	40	0.20	BASIC	0.2035	0.2131	0.2255
			FTE1	0.1778	0.1762	0.1741
			FTE2	0.1708	0.1797	0.1915
8.0	40	0.40	BASIC	0.3799	0.3962	0.4170
			FTE1	0.3700	0.3693	0.3684
			FTE2	0.3662	0.3837	0.4065

recommending a second order auto regression scheme for this purpose. In this regard, the work of Tucker and Robinson with validation may be viewed as *forecasting by persistence*, and in light of the NPRDC work, can hardly be expected to perform well in an absolute sense. Their work does, however, provide some comparisons that can be useful in choosing among alternative techniques, and much can be learned from them.

The first measure considered is the Mean Squared Error (MSE) of forecasts. The shrinkage estimation theory provides that these values should be about unity if persistence, independence over time, was agreeable. That is, useful validations can be obtained without modeling the time dependence of the attrition process. Only for a very few cases does this hold (i.e., Warrant Officers, see Tucker). However, if one method of estimation consistently produces noticeably smaller MSE values than do the other methods, this would provide sufficient support for that method's continued development.

The second measure is designed to consider performance in the original scale and is patterned after a chi-square statistic

$$\sum_{i}^{k} \frac{(\text{actuals} - \text{forecasts})^2}{\text{forecasts}} \qquad (2.8)$$

If the k cells are independent, then equation 2.8 is approximately a chi-square random variable with k degrees of

freedom. Hence, the man value is about k and the standard deviation is $(2k)^{1/2}$. Once again, we generally do not see values this small, but use it for comparison purposes. As a side note, the value of k when using the MLE is generally smaller than that of the other estimators due to the omission of those cells with MLE equal to zero over all estimation years, i.e., zero forecasts. This provides and unnatural advantage for the MLE, which must be kept in mind when interpreting the results.

Two disturbing features have emerged from the exploratory analysis. Often a technique that performs well for some cases using the MSE measure is not comparably supported by the chi-square measure. It seems the two measures do not weight the common features equally, and further study is necessary to develop understanding of the results.

Discussion with personnel at NPRDC concerning these matters has led to the introduction of a third measure, which will be called the Mean Absolute Deviation (MAD). This value is computed in the original scale. Moreover, since the cost structure of over-forecasting is not symmetric, we separate the two parts into the average *overage* deviation given by:

$$\text{ODEV} = \frac{1}{k} \sum_{1}^{k} [\text{forecasts} - \text{actuals}]^{+} \qquad (2.9)$$

and average *underage* deviation

$$UDEV = \frac{1}{k} \sum_{1}^{k} [\text{actuals} - \text{forecasts}]^{+} \qquad (2.10)$$

where the "+" refers to the positive part, as computed separately. Finally,

$$MAD = ODEV + UDEV \qquad (2.11)$$

This measure also has shown that in most cases, the overage exceeds the underage value, which in turn may lead to forecasting the need for an excess number of Marine officers.

III. EMPIRICAL BAYES ESTIMATION

A. BACKGROUND

Although the studies of Tucker and Robinson showed improvement over the current methods, no clear procedure was established and no plans for implementation were drafted. Problem areas were identified while working with small cell inventories and low loss rates. In the transformed scale, the James-Stein assumption of normal distribution of the observations with equal variance did not hold well in such cases. An alternate method of loss rate estimation is required to deal with this special case.

B. INTRODUCTION

Empirical Bayes is simply a generalization of the James-Stein estimator for use with small cell inventories and unstable variance. Random variables are assumed to be of the form:

$$X_i \sim N(\theta_i, d_i), \quad i=1,2,\ldots,k \qquad (3.1)$$

where θ_i are the unknown parameters to be estimated and d_i are known variances. The Bayesian assumption is that the θ_i's are themselves from some distribution, in this case also from the normal distribution with mean, ν, and variance, $\rho_i a$. The proportionality constant ρ_i is normalized so that

$$\sum \rho_i = k, \quad i=1,2,\ldots,k \tag{3.2}$$

where k is the number of observations or cells. Thus,

$$\theta_i \sim N(\nu, \rho_i a), \quad i=1,2,\ldots,k. \tag{3.3}$$

This distribution is known as the *prior* distribution of θ_i and represents the additional assumed information on the value of θ before observing X whose distribution depends on θ [Ref. 8:p. 553]. The use of this information produces an estimate which is a weighted average of the prior mean, ν, and the sample estimate x. In the empirical Bayes context, the values of ν and a, are also estimated from the data since all the information necessary is contained in the marginal distribution of X_i (unconditional on θ_i) [Ref. 6:p. 83]. This chapter focuses on that method.

For a more detailed explanation of the Bayesian family of estimators, the reader is referred to Casella [Ref. 6], Carter and Rolph [Ref. 4] and Efron and Morris [Ref. 9].

C. THE ESTIMATION METHOD

Fay and Herriot [Ref. 10] discuss the empirical Bayes technique and provide an estimate for the prior mean as

$$\nu^* = \frac{\sum_i x_i/(a^* + d_i)}{\sum_i 1/(a^* + d_i)}, \quad i=1,2,\ldots,k \tag{3.4}$$

where a^* is an unbiased estimate of a. The empirical Bayes estimate of θ, commonly called the mean of the *posterior*

where a^* is an unbiased estimate of a. The empirical Bayes estimate of Θ, commonly called the mean of the *posterior* distribution becomes

$$\delta^{EB} = \left(\frac{a^*}{(a^* + d_i)}\right)x_i + \left(\frac{d_i}{(a^* + d_i)}\right)\nu^*, \quad i=1,2,\ldots,k \qquad (3.5)$$

This is the mean of the conditional distribution of Θ given x [Ref. 8:p. 556]. The weights of each of the prior and sample means are determined by the prior and observed variances.

As a measure of the worth of this estimator, when the number of x's is ≥ 4 [Ref. 11], the *risk*, or sum of squared differences between the true unknown parameter and the empirical Bayes estimate, is less than that using the observed x's, for all θ_i. Efron and Morris [Ref. 9] provide a rigorous proof.

An important result of the empirical Bayes estimator is that shrinkage values are no longer constant, but depend upon information contained in each cell. Additionally, the point toward which the cell values are shrunk is the weighted mean, not the grand mean of the observations. Appendix A provides the derivation of the empirical Bayes estimator, variable shrinkage factors and calculation of the prior variance used in this study as they pertain to the Marine Corps officer attrition rates.

IV. VALIDATION PROCEDURES AND RESULTS

A. GENERAL

The outputs displayed in this chapter are excerpts from that produced by the Marine Corps Personnel Inventory Attrition Analysis (MCIAAP) (see Appendix B). Initially, comparisons will be made with the output displayed by Robinson to identify the validity of the Fortran code with the APL code used by Robinson, then new approaches will be displayed in an attempt to improve the results. The Fortran code in the MCIAAP incorporates most of the procedures studied by Tucker and Robinson. Specifically, the MLE, Transformed Scale Cell Average (TSCA) and the James-Stein (JS) estimators, shrinkage procedures and a basic inversion formula. The FTE inversion formula, the new measures of effectiveness from NPRDC, are also included. For the MAD measure of effectiveness, only maximum likelihood and the modified empirical Bayes variables will be presented, since these showed the most promise over all. All other variables are displayed by the MCIAAP if interest is warranted by the reader. Chapter II describes all of these methods in some detail.

B. HISTORICAL COMPARISONS

Tables 4-9 were developed using the same aggregation scheme and format as Robinson [Ref. 2:pp. 39-41], although

not all of the variables are included (MLE, TSCA, and James Stein are the only variables duplicated in this paper). Additionally, Robinson presented the number of cells, as k values, used for evaluation of each estimate in the aggregate Figure of Merit (FOM) display. These k values are not repeated in the current FOM tables, but are included, as parenthetical values, and discussed in the new MAD measure of effectiveness tables. Since this is the case here, the reader is reminded that the apparent competitive performance of the MLE in the following cases is due, in part, to the possible advantage of having a smaller number of cells used in this MOE's computation.

Since the computations were identical for both studies for these tables, the output was expected to be nearly identical or some pattern discernable between the two. In very few cases was this found to be true, and after verifying that the algorithms coded both in the APL and Fortran versions were correct, we have no reasonable explanation for the discrepancies of the programs. It is suspected that the transfer of raw data into an APL workspace, accomplished by Tucker and Robinson, may be at variance with that into Fortran files, done by contractor.

1. Aviation Aggregate Case

In the case of aviation FOM's, displayed in Table 4, 1981 values for LtCol's in the transformed scale are identical to Robinson's [Ref. 2:p. 39] while the last two

Table 4. AVIATION FIGURES OF MERIT

	1981	1982	1983
	TRANSFORMED FOM		
1st Lt			
MLE	4.0863	10.6590	11.1338
TSCA	3.7420	10.3636	10.8711
JS	3.9694	10.5671	11.1626
MOD JS	4.7352	11.2615	12.1206
EB	4.0753	10.7015	11.3303
MOD EB	3.1833	5.0568	6.2870
Lt Col			
MLE	4.3664	8.5033	8.1506
TSCA	5.7768	10.2994	10.3353
JS	5.7369	10.2468	10.2972
MOD JS	5.7208	10.1865	10.2756
EB	5.8482	10.2298	10.3534
MOD EB	4.3301	3.9824	2.7186
	ORIGINAL FOM		
1st Lt			
MLE	21.0501	43.1546	53.0538
TSCA	56.4603	93.7810	98.4715
TSCAM	53.7884	90.5164	95.3329
JS	58.0150	95.2150	100.1664
JSM	55.3844	91.9406	97.0163
MOD JS	63.0345	99.8419	105.4897
MOD JSM	60.6413	96.5615	102.3172
EB	58.8824	96.2449	101.2482
EBM	56.2111	92.9699	98.0958
MOD EB	19.7683	38.4637	49.7393
MOD EBM	20.7987	36.3673	48.1659
Lt Col			
MLE	33.9315	59.4200	22.1156
TSCA	38.4478	57.0987	69.0641
TSCAM	35.0390	48.9374	59.6957
JS	37.8248	55.1913	67.1696
JSM	33.2016	46.9901	56.7135
MOD JS	38.0291	51.5325	63.3908
MOD JSM	34.5466	45.1554	52.8285
EB	40.3421	54.9389	67.7091
EBM	36.7498	46.9657	57.8567
MOD EB	40.2056	70.5381	60.5749
MOD EBM	185.7476	65.1553	168.3976

years are smaller. The 1stLt FOM's are larger for all three years. The modified empirical Bayes (Mod EB) values were best overall yet still may be too large (ideal values tend to be close to unity). It is interesting to note that in most cases, the risk values tend to increase with future forecasts, yet the Mod EB FOM decreases as future forecasts increase with time.

In the original scale, MLE tended to be smaller than Robinson's for all years, while TSCA and James-Stein (JS) were all larger. The FTE inversion worked best for all variables except for Mod EB in the first and third validation years. MLE and Mod EB worked best for 1stLt's and are approximately within the range of one standard deviation of the number of cells in the computation. For LtCol's, FTE inverted JS was best with MLE a competitive second, yet all were not in the desired range.

In the MAD measure, Table 5, results are mixed between future forecasts (no set pattern) and variables for the two ranks (FTE Mod EB was best for 1stLt's and MLE was best LtCol's). The mix in overage and underage is most desirable in the first year for all cases, yet the number of cells (k) forecasted as overage exceed the underage forecast cell numbers in most cases. This is cause for concern and the performance of these estimates in this aggregate is suspect. The second year values for LtCol estimates displays a preferred ratio of k values, but the total MOE value may still be too large.

Table 5. MEAN ABSOLUTE DEVIATION (AVIATION)

	1981	1982	1983
1st Lt			
MLE			
UNDERAGE	0.1256(3)	0.0439(1)	0.3069(1)
OVERAGE	2.3985(8)	3.9267(10)	4.2522(10)
TOTAL	2.5240(11)	3.9706(11)	4.5591(11)
MOD EB			
UNDERAGE	0.1822(4)	0.0410(1)	0.3281(1)
OVERAGE	2.3625(8)	3.9040(11)	4.2351(11)
TOTAL	2.5447(12)	3.9451(12)	4.5633(12)
MOD EBM			
UNDERAGE	0.2518(5)	0.0580(2)	0.3420(1)
OVERAGE	2.1654(7)	3.6427(10)	3.9677(11)
TOTAL	2.4172(12)	3.7007(12)	4.3098(12)
Lt Col			
MLE			
UNDERAGE	0.6005(5)	1.5139(10)	0.0396(1)
OVERAGE	0.7925(8)	0.3928(3)	1.3467(12)
TOTAL	1.3930(13)	1.9067(13)	1.3863(13)
MOD EB			
UNDERAGE	0.6053(6)	1.4717(9)	0.0841(2)
OVERAGE	1.3083(10)	1.0970(8)	2.2331(14)
TOTAL	1.9136(16)	2.5687(17)	2.3173(16)
MOD EBM			
UNDERAGE	0.6890(6)	1.5366(10)	0.1167(2)
OVERAGE	1.0120(10)	0.7179(7)	1.7501(14)
TOTAL	1.7010(16)	2.2546(17)	1.8669(16)

2. Combat Support Aggregate Case

Table 6 displays the combat support aggregate output from the program. In the transformed scale comparison with Robinson [Ref. 2:p. 40], the values for the two ranks take opposite directions. Where the 1stLt FOM values are all

Table 6. COMBAT SUPPORT FIGURES OF MERIT

	1981	1982	1983
	TRANSFORMED FOM		
1st Lt			
MLE	2.1877	2.9028	2.2447
TSCA	1.6329	2.4580	1.8696
JS	1.6925	2.4816	1.8987
MOD JS	2.3684	3.0520	2.5356
EB	2.3323	3.1991	2.6868
MOD EB	2.2293	2.0048	2.1531
Lt Col			
MLE	1.3103	1.6978	2.3108
TSCA	0.7502	1.4070	1.7540
JS	0.6421	1.2345	1.5429
MOD JS	0.7009	1.0929	1.3082
EB	0.7736	1.2589	1.5799
MOD EB	1.8691	2.1236	4.9017
	ORIGINAL FOM		
1st Lt			
MLE	111.4260	112.1917	70.2767
TSCA	73.5008	90.0523	73.9380
TSCAM	79.0947	93.8244	80.9764
JS	76.9945	89.8064	71.5890
JSM	429.8503	84.4175	65.2639
MOD JS	143.1700	113.0083	90.6783
MOD JSM	168.9385	114.7083	92.6599
EB	102.9525	119.0900	96.3373
EBM	108.3578	111.2701	86.8138
MOD EB	68.7251	63.9054	48.2719
MOD EBM	176.3961	59.5760	43.0037
Lt Col			
MLE	36.0152	66.0261	42.4097
TSCA	61.7769	52.0346	99.7917
TSCAM	65.9535	49.6307	107.1016
JS	61.0511	49.2280	96.7861
JSM	66.3509	47.8490	118.2564
MOD JS	66.5470	49.9366	99.2566
MOD JSM	81.4722	50.2637	103.8460
EB	69.7508	55.9812	104.5311
EBM	71.5728	50.0281	109.2012
MOD EB	62.6316	50.3125	96.8693
MOD EBM	83.1337	48.6576	127.1541

larger (except for MLE in 1981),the LtCol values are smaller. Overall, all estimates are competitive with each other over all validation years with TSCA performing best for the 1stLt's and James-Stein outperforming the rest in the LtCol's. Note here also that the Mod EB estimate did not fair well in either case, and was the worst estimate in risk for the LtCol's.

In the original scale , a different and more random effect is noticed between Table 6 and the results of Robinson. Risk values tended to fluctuate with forecast lead time between estimates for 1stLt's where the LtCol values were more stable, yet were quite different from the transformed scale pattern. In this case, Mod EB performed well for 1stLt's, and was the only one within the desired range of standard deviations. MLE for the LtCol's was also the only estimate desirable for this measure. The Miller inverse showed poorly in this aggregate for all estimation methods, in every case exceeding it's counterpart.

In the MAD measure of effectiveness, Table 7, MLE performed the best in both ranks. The Mod EB effect is competitive in the 1stLt's case, and all variables in the LtCol case are closely grouped. The number of cells are heavily weighted towards an overage forecast in all three validation years, throughout this aggregate. Once again, this is not a desirable quality for a MOE.

Table 7. MEAN ABSOLUTE DEVIATION (COMBAT SUPPORT)

	1981	1982	1983
1st Lt			
MLE			
UNDERAGE	0.3737(14)	0.1617(8)	0.1781(8)
OVERAGE	0.6731(19)	0.9574(25)	0.8835(25)
TOTAL	1.0468(33)	1.1191(33)	1.0616(33)
MOD EB			
UNDERAGE	0.4028(15)	0.1275(9)	0.1427(9)
OVERAGE	0.7311(40)	0.9972(45)	0.9422(38)
TOTAL	1.1339(55)	1.1246(54)	1.0849(47)
MOD EBM			
UNDERAGE	0.4592(16)	0.1603(9)	0.1856(10)
OVERAGE	0.5930(39)	0.8318(45)	0.7575(37)
TOTAL	1.0522(55)	0.9921(54)	0.9431(47)
Lt Col			
MLE			
UNDERAGE	0.1360(9)	0.1688(7)	0.0541(5)
OVERAGE	0.1510(14)	0.2591(16)	0.3415(18)
TOTAL	0.2870(23)	0.4280(23)	0.3957(23)
MOD EB			
UNDERAGE	0.1638(11)	0.1721(7)	0.1162(6)
OVERAGE	0.2232(29)	0.3606(35)	0.4548(29)
TOTAL	0.3870(40)	0.5327(42)	0.5710(35)
MOD EBM			
UNDERAGE	0.2044(11)	0.1901(7)	0.1408(6)
OVERAGE	0.1263(29)	0.2240(35)	0.2589(29)
TOTAL	0.3307(40)	0.4142(42)	0.3996(35)

3. Ground Combat Aggregate Case

In the final aggregate studied by Robinson [Ref. 2:p. 41], that of ground combat MOS's, the results again, as displayed in Table 8, were mixed between computational methods. In the original scale, value differences tended to be larger than Robinson's in all cases for 1stLt's, showing a proportional growth over time. Conversely, LtCol values were smaller in the first two validation years and slightly larger in the final year. TSCA looks good for the 1stLt's with the others being competitive, yet the size of the values may not be desirable. Mod EB turned out the best for LtCol's, but all other estimates are competitive and in the desired range.

On the original scale, all values for the estimates for 1stLt's are too large, with Mod EB performing well in that regard. LtCol values are all larger than Robinson's, but the estimates are all within range of one standard deviation and, therefore, competitive. MLE, once again, shows well for this rank. In opposing fashion to that of the combat support aggregate, the FTE inversion outperformed all Basic inversions for all variables.

The MAD measure of effectiveness, Table 9 confirms the results found above in that Mod EB was best for 1stLt's and MLE did well for LtCol's. The values for 1stLt were found to be too large overall, while all estimates for LtCol were competitive and in the desired range. The mix between

Table 8. GROUND COMBAT FIGURES OF MERIT

	1981	1982	1983
	TRANSFORMED FOM		
1st Lt			
MLE	2.5764	4.4136	6.8744
TSCA	1.9741	3.8072	6.1152
JS	2.1737	3.9738	6.3970
MOD JS	3.0859	4.7919	7.5619
EB	2.5558	4.3626	6.9607
MOD EB	3.1538	3.5202	4.7069
Lt Col			
MLE	1.2126	1.9674	3.1084
TSCA	1.5715	2.3553	3.5566
JS	1.5571	2.2719	3.4365
MOD JS	1.6227	2.1317	3.1883
EB	1.7087	2.2464	3.3345
MOD EB	1.1964	2.5830	2.5029
	ORIGINAL FOM		
1st Lt			
MLE	87.3362	119.2637	143.9128
TSCA	87.9712	116.6767	145.6236
TSCAM	83.7648	111.3995	142.0902
JS	95.4717	122.9425	152.7354
JSM	92.0238	117.5264	149.3965
MOD JS	124.6490	147.6611	180.5502
MOD JSM	134.9472	142.5461	178.6173
EB	109.6349	136.6321	167.3234
EBM	105.5708	131.0400	163.8189
MOD EB	64.3647	81.2071	101.9511
MOD EBM	63.6623	77.8159	100.7150
Lt Col			
MLE	35.0538	45.8172	53.0111
TSCA	55.9649	198.8659	123.1306
TSCAM	43.5234	201.9953	122.8648
JS	56.3346	198.9037	122.2786
JSM	43.9957	201.3299	115.7143
MOD JS	60.0730	202.2959	122.9959
MOD JSM	52.3940	210.7551	121.4254
EB	62.7441	205.4249	126.6544
EBM	50.4049	209.3853	125.7642
MOD EB	48.0871	197.7316	111.9375
MOD EBM	38.1231	323.5654	242.4398

over-estimates and under-estimates, i.e., the number of cells forecasted over or under the actuals, is not a desirable one, yet the same mix holds true for all validation years. This also was the finding of both preceding aggregates.

Table 9. MEAN ABSOLUTE DEVIATION (GROUND COMBAT)

	1981	1982	1983
1st Lt			
MLE			
UNDERAGE	0.1525(5)	0.1855(10)	0.2466(10)
OVERAGE	1.9322(30)	2.5979(25)	3.1691(25)
TOTAL	2.0847(35)	2.7834(35)	3.4157(35)
MOD EB			
UNDERAGE	0.2340(10)	0.1323(8)	0.2220(8)
OVERAGE	1.7743(33)	2.3086(32)	2.8449(30)
TOTAL	2.0083(43)	2.4409(40)	3.0669(38)
MOD EBM			
UNDERAGE	0.2618(10)	0.1705(10)	0.2617(10)
OVERAGE	1.6492(33)	2.1738(30)	2.6935(28)
TOTAL	1.9111(43)	2.3443(40)	2.9552(38)
Lt Col			
MLE			
UNDERAGE	0.2313(10)	0.3034(11)	0.1368(10)
OVERAGE	0.4182(25)	0.4202(24)	0.6453(25)
TOTAL	0.6495(35)	0.7236(35)	0.7821(35)
MOD EB			
UNDERAGE	0.2315(11)	0.3299(14)	0.1426(12)
OVERAGE	0.6392(34)	0.6922(32)	0.9523(31)
TOTAL	0.8706(45)	1.0221(46)	1.0950(43)
MOD EBM			
UNDERAGE	0.2650(12)	0.3790(15)	0.1864(12)
OVERAGE	0.4071(33)	0.4427(31)	0.6535(31)
TOTAL	0.6720(45)	0.8217(46)	0.8399(43)

4. Test Aggregate Case

In an effort to experiment with aggregate groupings, another case was built upon which all estimation methods were applied, and the results compared with previously studied groups. The grade was held fixed, in this case the grade of Captain was used, with a LOS in the eighth year. Several MOS's were chosen so that to insure an aggregate with non-zero inventories. In the transformed scale, Table 10, all estimates have small and competitive values, with Mod EB and MLE doing well. In the original scale, once again, all estimates are well within range, but Mod EB and MLE were the standouts. The FTE inversion option did not fair well with this aggregate. All estimates in the MAD measure of effectiveness, Table 11, are very competitive, and the first validation year mix (overage and underage) may be the best yet. It is interesting to note that the cell split in the first validation year between overage and underage forecasts is contrary to that of the historical displays above. Although the total number of cells used (k=6) is much smaller than the previous table, it is felt that the aggregate scheme was the major factor in this finding, and the results are promising.

Table 10. TEST AGGREGATE FIGURES OF MERIT

	1981	1982	1983
	TRANSFORMED FOM		
Captain			
MLE	0.9603	2.3463	2.0842
TSCA	1.4151	3.3033	2.9152
JS	1.3753	3.3969	2.8725
MOD JS	1.3943	3.8165	2.8829
EB	1.6840	4.2698	3.2957
MOD EB	0.8249	1.7132	1.2980
	ORIGINAL FOM		
Captain			
MLE	3.0402	4.8104	4.9399
TSCA	4.6161	10.2779	7.5714
TSCAM	8.1467	7.6241	8.8347
JS	4.3970	10.3000	7.4726
JSM	7.3199	7.6666	8.3860
MOD JS	4.2990	10.7640	7.6371
MOD JSM	6.0079	8.3070	7.9259
EB	5.3660	12.0337	8.8679
EBM	6.7181	9.3915	8.7959
MOD EB	1.4463	6.0323	3.9066
MOD EBM	4.6465	4.3354	5.9175

Table 11. MEAN ABSOLUTE DEVIATION (TEST CASE)

	1981	1982	1983
Captain			
MLE			
UNDERAGE	0.2625(3)	0.0463(1)	0.2097(2)
OVERAGE	0.1351(3)	0.7374(5)	0.4391(4)
TOTAL	0.3976(6)	0.7837(6)	0.6488(6)
MOD EB			
UNDERAGE	0.1342(4)	0.0000(0)	0.0965(1)
OVERAGE	0.2575(2)	0.9601(6)	0.6306(5)
TOTAL	0.3918(6)	0.9601(6)	0.7270(6)
MOD EBM			
UNDERAGE	0.3712(4)	0.0809(1)	0.2352(2)
OVERAGE	0.0852(2)	0.6148(5)	0.3025(4)
TOTAL	0.4563(6)	0.6957(6)	0.5377(6)

V. CONCLUSIONS AND RECOMMENDATIONS

A. RESULTS

Omitting the fact of differences in computational methods between this paper and the Robinson thesis, it is clear that the results are still inconclusive at this time. In the historical comparisons, the estimates studied varied in their performance from one aggregate to another. The different risk calculations, in the transformed and original scales, as well as the different measures of effectiveness showed little agreement in estimation methods. Long range estimation still remains a problem since with time, the estimate risks increase significantly. Much emphasis was put on the modified empirical Bayes estimator in this paper and those specific results look promising under possibly different and more stable aggregate conditions.

One area of mention here concerning the results has to do with the composition of inventories used in the selected aggregates. It was noticed that the range of inventory (smallest to largest value in any particular MOS) was quite large, and may be affecting the results. In fact, Carter and Rolph [Ref. 4:p. 883] managed this point by further disaggregating into levels of activity as it concerned their work with fire alarm estimation.

B. CONCLUSIONS

The results of the previous feasibility studies and this paper leads to several conclusions. The chosen aggregate still seems to be the key element in the work of attrition estimation. It does seem clear that the aggregation scheme should be focused on fixed ranks, in narrow LOS ranges and multiple MOS's. This conclusion is based upon the relative success of the test aggregate utilized in this work, and upon similar success by Carter and Rolph. Additionally, the partitioning of inventory levels in the aggregate should stabilize the estimation method performance. All the estimation method studied to date have their strengths, but still, no one method of estimation excels over the others.

C. RECOMMENDATIONS

No estimation method studied in this paper is recommended for implementation at this time. In addition, no method is recommended for removal from consideration since basic aggregation problems are felt to be the largest contributory factor to the mixed results found here. These recommendations are presented for further study.

1. The aggregation method offered by Captain Larsen [Ref. 3] should be examined further with its relevance to the scheme in the test case used here.

2. Utilization of the detailed data provided by NPRDC as soon as possible in order to expand the available information and open new avenues of aggregation.

3. Re-examine the inventory mean value used in the FTE inverse to determine its future role in the estimation process.

APPENDIX A. ESTIMATION ALGORITHMS

A. NOTATION

The following notation is a generalization of that used by Robinson [Ref. 2:p. 71] as it pertains to the aggregation scheme used in this paper. The indexing system has been changed to reflect the use of officer grades to identify the cells.

i = index of MOS cells in the aggregate.

j = index of LOS cells in the aggregate.

k = index of Grade cells in the aggregate.

t = index of time periods in the aggregate.

inv(i,j,k,t) = inventory with MOS i, LOS j and Grade k during year t.

y(i,j,k,t) = number of attritions in cell (i,j,k) during year t.

n(i,j,k,t) = maximum {y(i,j,k,t),.5[inv(i,j,k,t) + inv(i,j,k,t + 1)]}

D = matrix identifying cells with no inventory over all estimation years.

d(i,j,k) = 0, if cell is a structural zero.

d(i,j,k) = 1, if cell is not a structural zero.

B. FREEMAN-TUKEY EXACT INVERSION ALGORITHM

This algorithm provides a formula for the inverse of the Freeman-Tukey double arcsine transformation to express the means of the transformed values into proportions on the original scale.

STEP 1. Transform the data using the Freeman-Tukey double arcsine transformation.

$$x(i,j,k,t) = 0.5[n(i,j,k,t)+0.5]^{1/2}\{\sin^{-1}[\frac{2(y(i,j,k,t))}{(n(i,j,k,t)+1)} -1] + \sin^{-1}[\frac{2(y(i,j,k,t)+1}{(n(i,j,k,t)+1)} -1]\} \quad (A.1)$$

STEP 2. Calculate the average cell inventory and transformed values over all T.

$$\bar{x}(i,j,k) = (1/T)\sum_{t} x(i,j,k), \quad (A.2)$$

$$\overline{inv}(i,j,k) = (1/T)\sum_{t} inv(i,j,k). \quad (A.3)$$

STEP 3. Check for inversion values less than zero or greater than one. The algorithm is terminated at this point if these conditions are met.

$$tm = \bar{x}(i,j,k)/(\overline{inv}(i,j,k)+0.5)^{1/2} + \pi/2, \quad (A.4)$$

$$tm_1 = \sin^{-1}(1/(\overline{inv}(i,j,k) + 1))^{1/2}, \quad (A.5)$$

If $tm < tm_1$, FTE = 0.0,

IF $tm > (\pi - tm_1)$, FTE = 1.0,

STEP 4. Compute the sgn function.

If cos(tm) > 0.0 sgn = 1.0,

If cos(tm) < 0.0 sgn = -1.0.

STEP 5. Computer the inverted value of the transformed variable

$$FTE = 0.5(1-sgn\{1-[\sin(tm) + \frac{\sin(tm) - 1/\sin(tm)}{\overline{inv}(i,j,k)}]^2\}^{1/2}\} \quad (A.6)$$

C. TRANSFORM SCALE VARIANCE ALGORITHM

This algorithm produces the variance of the transformed data within the unstable variance range where the assumption of normality does not hold. As discussed in Chapter II, this range of instability was determined to be between 1.001 and 2.2. The method for computing the variance within this range is presented here.

STEP 1. Compute the transformed scale values to be used for the variance. Since x(i,j,k) was computed in the FTE inversion algorithm of this appendix, the formulae will not be repeated.

$$\bar{x}(t) = \sum_{ijk} [0.5 + inv(i,j,k,t)]^{1/2}, \quad (A.7)$$

$$z(i,j,k) = \bar{x}(i,j,k) + \frac{x(t)}{T}(\pi/2). \quad (A.8)$$

STEP 2. Compute the variance of transformed cell values. The formula is derived in Chapter II as a linear regression equation. The range of the variance is limited to $0.05 \leq var(i,j,k) \leq 1.0$.

$$var(i,j,k) = az(i,j,k)^{b1}(z(i,j,k) - 1)^{b2}, \quad 1.001 \leq z(i,j,k) \leq 2.2 \qquad (A.9)$$

where a = 1.6835, b1 = -0.8934, and b2 = 0.8991.

D. EMPIRICAL BAYES SHRINKAGE ALGORITHM

This algorithm computes the empirical Bayes shrinkage values and estimator of attrition rates. Prior to entering this algorithm, the value for a has been set to 0.0.

STEP 1. Initialization.

$$ai = a. \qquad (A.10)$$

STEP 2. Define the α and γ parameters in the empirical Bayes context.

$$\alpha = \sum_{ijk} 1/(a + var(i,j,k)), \qquad (A.11)$$

where var(i,j,k) is defined by formula (A.9).

$$\gamma = \sum_{ijk} \alpha / \sum_{ijk} 1/(\alpha + var(i,j,k)). \qquad (A.12)$$

STEP 3. Compute the empirical Bayes prior variance.

$$a = a - \frac{k-1-\sum_{ijk} \alpha[\bar{x}(i,j,k)-\bar{z}]^2}{\sum_{ijk} [\alpha(\bar{x}(i,j,k) - \bar{z}]^2}, \qquad (A.13)$$

where $z = \sum_{ijk} \gamma\bar{x}(i,j,k)$ and $k = \sum_{ijk} d(i,j,k)$.

STEP 4. Check for the following conditions and branch accordingly.

If, $a < 0.0$ set $a = 0.0$ and go to step 5.

If, $|a - ai| > .0001$ return to Step 1.

STEP 5. Compute the empirical Bayes estimator.

$$\bar{x}^{EB}(i,j,k) = [\frac{a}{a+var(i,j,k)}]\bar{x}(i,j,k) + \qquad (A.14)$$

$$[\frac{var(i,j,k)}{a+var(i,j,k)}]\bar{z}.$$

STEP 6. Compute the empirical Bayes shrinkage set.

$$shr^{EB}(i,j,k) = var(i,j,k)/(a + var(i,j,k)). \qquad (A.15)$$

E. MODIFIED EMPIRICAL BAYES SHRINKAGE ALGORITHM

This algorithm differs only slightly from that of the empirical Bayes shrinkage algorithm listed above, and for

that reason, only the pertinent steps will be displayed here.

Consider a particular cell and its transformed value x(t) for t=1,...,T in the estimation set. To form a weighted time average, set

$$xt(t) = x(t)/[0.5 + inv(t)]^{1/2} \qquad (A.16)$$

This leads to a modification of step 1 in the following form.

STEP 1. Compute the weighted transformed scale average.

$$\overline{xt} = \frac{1}{T} \sum_{t} xt(t) \qquad (A.17)$$

Since the weights will modify the variance of the weighted transformed scale values.

$$var[xt(t)] = var(i,j,k)/(0.5 + inv(t)) \qquad (A.18)$$

$$var[\overline{xt}(t)] = (\frac{1}{T})^2 \sum_{t} var[xt(t)] = vt/T \qquad (A.19)$$

The extreme right of formula A.19 serves to define vt, the approximation for the weighted variance in the validation set for that cell.

The values of $\overline{xt}$ and $var(\overline{xt})$ are carried throughout this version of empirical Bayes estimation to form the values XTEB for the empirical Bayes attrition rate. For the purposes of inversion to the original scale, the usual Basic

formula is modified since the weighting factor, 0.5 + inv, is already incorporated into the transform value. The new Basic inversion for this version is simply as follows:

$$p = 1/2[1 + \sin(XTEB)] \qquad (A.20)$$

APPENDIX B. MARINE CORPS PERSONNEL INVENTORY ATTRITION ANALYSIS PROGRAM

```
      PROGRAM MARCOR                                                    MAR00010
*****************************************************************     MAR00020
*     MARINE CORPS PERSONNEL INVENTORY ANALYSIS PROGRAM           *     MAR00030
*                                                                 *     MAR00040
*     PURPOSE: TO PROVIDE FUTURE PERSONNEL LOSS ESTIMATION OF THE *     MAR00050
*              MARINE CORPS OFFICER STRUCTURE BY SEVERAL DIFFERENT*     MAR00060
*              ESTIMATION SCHEMES.                                *     MAR00070
*     PROGRAMMERS: LUIS URIBE, INDEPENDENT CONTRACTOR             *     MAR00080
*                  CAPT C R DICKINSON, USMC                       *     MAR00090
*     DESCRIPTION: WRITTEN IN FORTRAN 77 FOR THE IBM 3033 MAINFRAME*    MAR00100
*              COMPUTER RESIDENT AT NPS, THIS PROGRAM INCORPORATES*     MAR00110
*              METHODS EXPLORED BY MAJOR D.D. TUCKER , MAJOR J.R. *     MAR00120
*              ROBINSON AND CAPTAIN DICKINSON. ESTIMATION SCHEMES *     MAR00130
*              INCLUDED ARE:                                      *     MAR00140
*                           MAXIMUM LIKELIHOOD ESTIMATION (MLE)   *     MAR00150
*                           TRANSFORMED SCALE CELL AVERAGE (TSCA) *     MAR00160
*                           JAMES-STEIN                           *     MAR00170
*                           LIMITED TRANSLATION JAMES-STEIN       *     MAR00180
*                           EMPIRICAL BAYES ESTIMATION            *     MAR00190
*              RISK ANALYSIS WAS PERFORMED IN BOTH THE ORIGINAL AND*     MAR00200
*              TRANSFORMED SCALES. ON THE TRANSFORMED SCALE,      *     MAR00210
*              ACHIEVED BY USING THE FREEMAN-TUKEY DOUBLE ARCSINE *     MAR00220
*              TRANSFORMATION, THE RISK WAS DEFINED BY THE AVERAGE*     MAR00230
*              SQUARED DEVIATION OF ACTUAL AND ESTIMATED VALUES.  *     MAR00240
*              FOR THE ORIGINAL SCALE, THE TRANSFORMED VALUES WERE*     MAR00250
*              FIRST INVERTED USING TWO TECHNIQUES. AN AD-HOC     *     MAR00260
*              FORMULA USED BY TUCKER AND ROBINSON AND AN EXACT   *     MAR00270
*              INVERSE FORMULA FROM JOHN J MILLER. TWO MEASURES OF*     MAR00280
*              EFFECTIVENESS, CHI-SQUARE GOODNESS OF FIT AND MEAN *     MAR00290
*              ABSOLUTE DEVIATION (MAD), WERE THEN EVALUATED.     *     MAR00300
*              RESULTS ARE PRINTED, WITH ALL ARRAYS COMPUTED, AT THE*    MAR00310
*              COMPLETION OF THE PROGRAM.                         *     MAR00320
*     VARIABLE DESCRIPTION:                                       *     MAR00330
*              THE FOLLOWING LIST OF VARIABLES, WITH A BRIEF      *     MAR00340
*              DESCRIPTION, FORM THE BASIS FOR THE VARIABLE NAMES *     MAR00350
*              FOR THE ENTIRE PROGRAM.                            *     MAR00360
*                    MLE - MAXIMUM LIKELIHOOD ESTIMATOR           *     MAR00370
*                   TSCA - TRANSFORMED SCALE CELL AVERAGE         *     MAR00380
*                      J - JAMES-STEIN ESTIMATOR                  *     MAR00390
*                     J1 - JAMES-STEIN ESTIMATOR MODIFIED BY      *     MAR00400
*                          A VARIATION ON SHRINKAGE               *     MAR00410
*                    JLT - LIMITED TRANSLATION JAMES-STEIN        *     MAR00420
*                     EB - EMPIRICAL BAYES                        *     MAR00430
*                    TEB - EMPIRICAL BAYES MODIFIED BY A          *     MAR00440
*                          VARIATION ON                           *     MAR00450
*        TRANSFORMED SPACE VARIABLES                              *     MAR00460
*              THE FOLLOWING ARE UNIQUE VARIABLES IN THE TRANSFORMED*    MAR00470
*              SPACE:                                             *     MAR00480
*                     X - VALUE OF THE FREEMAN-TUKEY              *     MAR00490
```

```
*                    TRANSFORMATION                                  *  MAR00500
*                XB - (X-BAR) AVERAGE TRANSFORM VALUE                *  MAR00510
*                     OVER ESTIMATION YEARS                          *  MAR00520
*               XBB - GRAND MEAN OF TRANSFORMED VALUES               *  MAR00530
*               SSE - SUM OF SQUARED DIFFERENCES, ERROR              *  MAR00540
*                     (X AND X-BAR)                                  *  MAR00550
*               SST - SUM OF SQUARED DIFFERENCES, TOTAL              *  MAR00560
*                     (X AND XBB)                                    *  MAR00570
*               SSB - SUM OF SQUARED DIFFERENCES, BETWEEN            *  MAR00580
*                           (SST-SSE)                                *  MAR00590
*                 SHR - JAMES-STEIN SHRINKAGE VALUES                 *  MAR00600
*                SHR1 - MODIFIED JAMES-STEIN SHRINKAGE               *  MAR00610
*                       VALUES                                       *  MAR00620
*              SHREB - EMPIRICAL BAYES SHRINKAGE VALUES              *  MAR00630
*             SHRTEB - MODIFIED EMPIRICAL BAYES SHRINKAGE            *  MAR00640
*                       VALUES                                       *  MAR00650
*         THE FOLLOWING ARE MODIFIED BASIC VARIABLES IN THE          *  MAR00660
*         TRANSFORMED SPACE. THIS IS DONE BY PLACING AN 'X' FOR      *  MAR00670
*         A PURE TRANSFORMED VARIABLE, OR AN 'R' FOR A RISK          *  MAR00680
*         VARIABLE IN FRONT OF THE BASIC VARIABLE.                   *  MAR00690
*                 MLE             RML (MLE)                          *  MAR00700
*                 TSCA            RTSCA                              *  MAR00710
*                 XJ              RJ                                 *  MAR00720
*                 XJ1             RJ1                                *  MAR00730
*                 XJLT            RJLT                               *  MAR00740
*                 XEB             RXEB                               *  MAR00750
*                 XTEB            RTEB                               *  MAR00760
*    ORIGINAL SPACE VARIABLES                                        *  MAR00770
*         THE FOLLOWING ARE MODIFIED BASIC VARIABLES IN THE          *  MAR00780
*         ORIGINAL SPACE. SIMILAR TO WHAT IS DONE IN THE             *  MAR00790
*         TRANSFORMED SPACE, A 'P' SIGNIFIES A PURE INVERTED         *  MAR00800
*         VARIABLE, AND AN 'RR' REPRESENTS A RISK VARIABLE. AN       *  MAR00810
*         'M' AT THE END OF A VARIABLE NAME IDENTIFIES A MILLER      *  MAR00820
*         INVERSE VARIABLE.                                          *  MAR00830
*                 PMLE            RRML (MLE)                         *  MAR00840
*                 PJ              RRJ                                *  MAR00850
*                 PJM             RRPJM                              *  MAR00860
*                 PJ1             RRJ1                               *  MAR00870
*                 PJ1M            RRJ1M                              *  MAR00880
*                 PJLT            RRJLT                              *  MAR00890
*                 PEB             RRXEB                              *  MAR00900
*                 PEBM            RRXEBM                             *  MAR00910
*                 PTEB            RRRTEB                             *  MAR00920
*                 PTEBM           RRTEBM                             *  MAR00930
*         THE FOLLOWING LIST IS OF PREFIXES USED TO DESCRIBE         *  MAR00940
*         THE MEAN ABSOLUTE DEVIATION VARIABLES:                     *  MAR00950
*                MO - MEAN DEVIATION 'OVERAGE'                       *  MAR00960
*                MU - MEAN DEVIATION 'UNDERAGE'                      *  MAR00970
*                MD - MEAN 'TOTAL DEVIATION                          *  MAR00980
*               KVO - NUMBER OF 'OVERAGE' ESTIMATES IN THE           *  MAR00990
*                     AGGREGATE                                      *  MAR01000
*               KVU - NUMBER OF 'UNDERAGE' ESTIMATES IN THE          *  MAR01010
*                     AGGREGATE                                      *  MAR01020
*               KVD - NUMBER OF 'TOTAL' ESTIMATION DEVIATIONS        *  MAR01030
*                     IN THE AGGREGATE                               *  MAR01040
*         THESE PREFIXES ARE APPENDED TO THE BASIC VARIABLES TO      *  MAR01050
```

```
*               FORM THE MAD VARIABLES AS FOLLOWS                          *  MAR01060
*                   MOTSCA MUTSCA MDTSCA KVOTSA KVUTSA KVDTSA (TSCA)       *  MAR01070
*                   MOTSAM MUTSAM MDTSAM KVOTSM KVUTSM KVDTSM (TSCAM)      *  MAR01080
*                   MOPMLE MUPMLE MDPMLE KVOPML KVUPML KVDPML (PMLE)       *  MAR01090
*                   MOPJ   MUPJ   MDPJ   KVOPJ  KVUPJ  KVDPJ  (PJ)        *  MAR01100
*                   MOPJM  MUPJM  MDPJM  KVOPJM KVUPJM KVDPJM (PJM)       *  MAR01110
*                   MOPJ1  MUPJ1  MDPJ1  KVOPJ1 KVUPJ1 KVDPJ1 (PJ1)       *  MAR01120
*                   MOPJ1M MUPJ1M MDPJ1M KVOP1M KVUP1M KVDP1M (PJ1M)      *  MAR01130
*                   MOPEB  MUPEB  MDPEB  KVOPEB KVUPEB KVDPEB (PEB)       *  MAR01140
*                   MOPEBM MUPEBM MDPEBM KVOPBM KVUPBM KVDPBM (PEBM)      *  MAR01150
*                   MOPTEB MUPTEB MDPTEB KVOPTB KVUPTB KVDPTB (PTEB)      *  MAR01160
*                   MOPTBM MUPTBM MDPTBM KVOBTM KVUBTM KVDBTM (PTEBM)     *  MAR01170
*       INPUT/OUTPUT:                                                      *  MAR01180
*               INPUT IS TAKEN FROM A FILE CREATED FROM A TAPE             *  MAR01190
*                   PROVIDED BY NPRDC                                      *  MAR01200
*               OUTPUT IS TO THE PRINTER                                   *  MAR01210
*********************************************************************************  MAR01220
**        FIXED PARAMETERS                                                     MAR01230
      PARAMETER (MXYR=7, MXLOS=31, MXGRD=10, MXMOS=15)                        MAR01240
**        INPUT PARAMETERS                                                     MAR01250
      INTEGER ST1, ST2, LYR                                                   MAR01260
      INTEGER SLOS1, SLOS2                                                    MAR01270
      INTEGER SMOS(MXMOS), NMOS                                               MAR01280
      INTEGER SGRD(MXGRD), NGRD                                               MAR01290
**        VARIABLES IN ORIGINAL SPACE                                          MAR01300
      INTEGER T, FLAG, IVYR(MXYR)                                             MAR01310
      INTEGER*2  D(MXMOS, MXLOS, MXGRD)                                       MAR01320
      INTEGER KV(MXYR)                                                        MAR01330
      REAL AVCINV,AINV,ATT,ATT1                                               MAR01340
      REAL CINV(MXMOS,MXLOS,MXGRD,MXYR),Y(MXMOS,MXLOS,MXGRD,MXYR)            MAR01350
      REAL ACINV(MXMOS,MXLOS,MXGRD)                                           MAR01360
      REAL PJ(MXMOS,MXLOS,MXGRD), PJM(MXMOS,MXLOS,MXGRD)                      MAR01370
      REAL PMLE(MXMOS,MXLOS,MXGRD), PJLT(MXMOS,MXLOS,MXGRD)                   MAR01380
      REAL PJ1(MXMOS,MXLOS,MXGRD), PJ1M(MXMOS,MXLOS,MXGRD)                    MAR01390
      REAL PEB(MXMOS,MXLOS,MXGRD), PEBM(MXMOS,MXLOS,MXGRD)                    MAR01400
      REAL PTEB(MXMOS,MXLOS,MXGRD), PTEBM(MXMOS,MXLOS,MXGRD)                  MAR01410
      REAL RRTSCA(MXYR),RRJ(MXYR),RRJLT(MXYR),RRML(MXYR)                      MAR01420
      REAL RRTSAM(MXYR),RRJM(MXYR),RRJ1(MXYR),RRJ1M(MXYR)                     MAR01430
      REAL RRPEB(MXYR),RRPEBM(MXYR)                                           MAR01440
      REAL RRPTEB(MXYR),RRPTBM(MXYR)                                          MAR01450
**        VARIABLES IN TRANSFORMED SPACE                                       MAR01460
      REAL X(MXMOS,MXLOS,MXGRD,MXYR)                                          MAR01470
      REAL XT(MXMOS,MXLOS,MXGRD,MXYR)                                         MAR01480
      REAL XB(MXMOS,MXLOS,MXGRD), TSCA(MXMOS,MXLOS,MXGRD)                     MAR01490
      REAL XTB(MXMOS,MXLOS,MXGRD)                                             MAR01500
      REAL XJ(MXMOS,MXLOS,MXGRD), XJLT(MXMOS,MXLOS,MXGRD)                     MAR01510
      REAL XEB(MXMOS,MXLOS,MXGRD), XJ1(MXMOS,MXLOS,MXGRD)                     MAR01520
      REAL XTEB(MXMOS,MXLOS,MXGRD)                                            MAR01530
      REAL XMLE(MXMOS,MXLOS,MXGRD)                                            MAR01540
      REAL SHREB(MXMOS,MXLOS,MXGRD)                                           MAR01550
      REAL SHRTEB(MXMOS,MXLOS,MXGRD)                                          MAR01560
      REAL RML(MXYR), RSL(MXYR),RXEB(MXYR)                                    MAR01570
      REAL RXTEB(MXYR)                                                        MAR01580
      REAL RTSCA(MXYR),RTJ(MXYR),RTJLT(MXYR)                                  MAR01590
      REAL RJ(MXYR),RJLT(MXYR), RJ1(MXYR)                                     MAR01600
      REAL TSCAM(MXMOS,MXLOS,MXGRD)                                           MAR01610
```

```
      REAL V(MXMOS,MXLOS,MXGRD)                                                 MAR01620
      REAL VT(MXMOS,MXLOS,MXGRD)                                                MAR01630
      REAL*8 SHR, SHRX, SHR1, SSB, SST, SSE, XBB                                MAR01640
**       VARIABLES FOR FIGURE OF MERIT                                          MAR01650
      INTEGER KVOTSA(MXYR),KVOTSM(MXYR),KVOPJ(MXYR),KVOPJ1(MXYR)                MAR01660
      INTEGER KVOPJM(MXYR),KVOPEB(MXYR),KVOP1M(MXYR),KVOPBM(MXYR)               MAR01670
      INTEGER KVOPTB(MXYR),KVOBTM(MXYR)                                         MAR01680
      INTEGER KVOPML(MXYR)                                                      MAR01690
      INTEGER KVUTSA(MXYR),KVUTSM(MXYR),KVUPJ(MXYR),KVUPJ1(MXYR)                MAR01700
      INTEGER KVUPJM(MXYR),KVUPEB(MXYR),KVUP1M(MXYR),KVUPBM(MXYR)               MAR01710
      INTEGER KVUPTB(MXYR),KVUBTM(MXYR)                                         MAR01720
      INTEGER KVUPML(MXYR)                                                      MAR01730
      INTEGER KVDTSA(MXYR),KVDTSM(MXYR),KVDPJ(MXYR),KVDPJ1(MXYR)                MAR01740
      INTEGER KVDPJM(MXYR),KVDPEB(MXYR),KVDP1M(MXYR),KVDPBM(MXYR)               MAR01750
      INTEGER KVDPTB(MXYR),KVDBTM(MXYR)                                         MAR01760
      INTEGER KVDPML(MXYR)                                                      MAR01770
      REAL MUTSCA(MXYR),MUTSAM(MXYR),MUPJ(MXYR),MUPJ1(MXYR),MUPJM(MXYR)        MAR01780
      REAL MUPJ1M(MXYR),MUPEBM(MXYR),MUPEB(MXYR),MUPMLE(MXYR)                   MAR01790
      REAL MUPTBM(MXYR),MUPTEB(MXYR)                                            MAR01800
      REAL MOTSCA(MXYR),MOTSAM(MXYR),MOPJ(MXYR),MOPJ1(MXYR),MOPJM(MXYR)        MAR01810
      REAL MOPJ1M(MXYR),MOPEBM(MXYR),MOPEB(MXYR),MOPMLE(MXYR)                   MAR01820
      REAL MOPTBM(MXYR),MOPTEB(MXYR)                                            MAR01830
      REAL MDTSCA(MXYR),MDTSAM(MXYR),MDPJ(MXYR),MDPJ1(MXYR),MDPJM(MXYR)        MAR01840
      REAL MDPJ1M(MXYR),MDPEBM(MXYR),MDPEB(MXYR),MDPMLE(MXYR)                   MAR01850
      REAL MDPTBM(MXYR),MDPTEB(MXYR)                                            MAR01860
      REAL MILLER                                                               MAR01870
**       INPUT DATA AREAS                                                       MAR01880
      INTEGER MOS, LOS, GRADE, YR, INV                                          MAR01890
      INTEGER MOS1,LOS1,GRADE1,YR1,INV1                                         MAR01900
**       INPUT INITIALIZATION                                                   MAR01910
      DATA KV/MXYR*0/                                                           MAR01920
      DATA MOPMLE/MXYR*0/,MUPMLE/MXYR*0/,MDPMLE/MXYR*0/                         MAR01930
      DATA KVOPML/MXYR*0/,KVUPML/MXYR*0/,KVDPML/MXYR*0/                         MAR01940
      DATA MOTSCA/MXYR*0/,MUTSCA/MXYR*0/,MDTSCA/MXYR*0/                         MAR01950
      DATA KVOTSA/MXYR*0/,KVUTSA/MXYR*0/,KVDTSA/MXYR*0/                         MAR01960
      DATA MOTSAM/MXYR*0/,MUTSAM/MXYR*0/,MDTSAM/MXYR*0/                         MAR01970
      DATA KVOTSM/MXYR*0/,KVUTSM/MXYR*0/,KVDTSM/MXYR*0/                         MAR01980
      DATA MOPJ/MXYR*0/,MUPJ/MXYR*0/,MDPJ/MXYR*0/                               MAR01990
      DATA KVOPJ/MXYR*0/,KVUPJ/MXYR*0/,KVDPJ/MXYR*0/                            MAR02000
      DATA MOPJM/MXYR*0/,MUPJM/MXYR*0/,MDPJM/MXYR*0/                            MAR02010
      DATA KVOPJM/MXYR*0/,KVUPJM/MXYR*0/,KVDPJM/MXYR*0/                         MAR02020
      DATA MOPJ1/MXYR*0/,MUPJ1/MXYR*0/,MDPJ1/MXYR*0/                            MAR02030
      DATA KVOPJ1/MXYR*0/,KVUPJ1/MXYR*0/,KVDPJ1/MXYR*0/                         MAR02040
      DATA MOPJ1M/MXYR*0/,MUPJ1M/MXYR*0/,MDPJ1M/MXYR*0/                         MAR02050
      DATA KVOP1M/MXYR*0/,KVUP1M/MXYR*0/,KVDP1M/MXYR*0/                         MAR02060
      DATA MOPEB/MXYR*0/,MUPEB/MXYR*0/,MDPEB/MXYR*0/                            MAR02070
      DATA KVOPEB/MXYR*0/,KVUPEB/MXYR*0/,KVDPEB/MXYR*0/                         MAR02080
      DATA MOPEBM/MXYR*0/,MUPEBM/MXYR*0/,MDPEBM/MXYR*0/                         MAR02090
      DATA KVOPBM/MXYR*0/,KVUPBM/MXYR*0/,KVDPBM/MXYR*0/                         MAR02100
      DATA MOPTEB/MXYR*0/,MUPTEB/MXYR*0/,MDPTEB/MXYR*0/                         MAR02110
      DATA KVOPTB/MXYR*0/,KVUPTB/MXYR*0/,KVDPTB/MXYR*0/                         MAR02120
      DATA MOPTBM/MXYR*0/,MUPTBM/MXYR*0/,MDPTBM/MXYR*0/                         MAR02130
      DATA KVOBTM/MXYR*0/,KVUBTM/MXYR*0/,KVDBTM/MXYR*0/                         MAR02140
*                                                                               MAR02150
*********************************************************************************MAR02160
```

```
**       READ PARAMETER DATA TO USE IN THIS RUN.                    **  MAR02170
*************************************************************************MAR02180
      WRITE(5,100)                                                       MAR02190
100   FORMAT(' ENTER 1ST YEAR AND LAST YEAR TO USE FOR ESTIMATION')      MAR02200
      READ(5,*) ST1,ST2                                                  MAR02210
      WRITE(6,*) 'ESTIMATION YEARS: ',ST1,ST2                            MAR02220
*                                                                        MAR02230
      WRITE(5,101)                                                       MAR02240
101   FORMAT(' ENTER LAST YEAR PRESENT IN THE DATA BASE')                MAR02250
      READ(5,*) LYR                                                      MAR02260
      WRITE(6,*) 'LAST YEAR AVAILABLE FOR VALIDATION:',LYR               MAR02270
*                                                                        MAR02280
      WRITE(5,102)                                                       MAR02290
102   FORMAT(' ENTER NO. OF MOS FOLLOWED BY ARRAY OF SAME LENGTH')       MAR02300
      READ(5,*) NMOS, (SMOS(I), I=1,NMOS)                                MAR02310
      WRITE(6,*) 'MOS SELECTED:', (SMOS(I), I=1,NMOS)                    MAR02320
*                                                                        MAR02330
      WRITE(5,103)                                                       MAR02340
103   FORMAT(' ENTER 1ST AND LAST LOS VALUE TO USE')                     MAR02350
      READ(5,*) SLOS1,SLOS2                                              MAR02360
      WRITE(6,*) 'LOS RANGE:',SLOS1,SLOS2                                MAR02370
*                                                                        MAR02380
      WRITE(5,104)                                                       MAR02390
104   FORMAT(' ENTER NO. OF GRADE FOLLOWED BY ARRAY OF SAME LENGTH')     MAR02400
      READ(5,*) NGRD, (SGRD(I), I=1,NGRD)                                MAR02410
      WRITE(6,*) 'GRADES SELECTED', (SGRD(I), I=1,NGRD)                  MAR02420
*                                                                        MAR02430
      WRITE(5,105)                                                       MAR02440
105   FORMAT(' ENTER DEE FACTOR')                                        MAR02450
      READ(5,*) DEE                                                      MAR02460
      WRITE(6,*) 'DEE FACTOR:',DEE                                       MAR02470
*                                                                        MAR02480
**       COMPUTE START AND END YEARS FOR VALIDATION                     MAR02490
      NYR=LYR-ST1+1                                                      MAR02500
      NVYR=LYR-ST2                                                       MAR02510
      NLOS=SLOS2-SLOS1+1                                                 MAR02520
      T=ST2-ST1+1                                                        MAR02530
      DO 41 I=1,NVYR                                                     MAR02540
       IVYR(I)=ST2+I                                                     MAR02550
41    CONTINUE                                                           MAR02560
*************************************************************************MAR02570
**       INITIALIZE D ARRAY TO ALLOW FINDING MISSING VALUES.        **  MAR02580
*************************************************************************MAR02590
      DO 1 I=1,MXMOS                                                     MAR02600
       DO 2 J=1,MXLOS                                                    MAR02610
        DO 3 K=1,MXGRD                                                   MAR02620
         D(I,J,K)=-9999                                                  MAR02630
3       CONTINUE                                                         MAR02640
2      CONTINUE                                                          MAR02650
1     CONTINUE                                                           MAR02660
*************************************************************************MAR02670
**       TAPE PROCESSING TO INPUT INVENTORY AND ATTRITION VALUES,   **  MAR02680
**       DEVELOPE D MATRIX ACCORDING TO EXTRACTION CRITERIA.        **  MAR02690
*************************************************************************MAR02700
      FLAG=0                                                             MAR02710
      IEOF=0                                                             MAR02720
```

```
      CALL READER(MOS1,LOS1,GRADE1,YR1,INV1,ATT1, NMOS,SMOS,          MAR02730
     *        NGRD,SGRD, SLOS1,SLOS2, ST1,ST2, IM1,IL1,IG1,IT1,IEOF)  MAR02740
*                                                                     MAR02750
      IF(IEOF .NE. 0) THEN                                           MAR02760
               WRITE(6,*) '**** NO DATA MEETS SELECTIONS REQS'       MAR02770
               STOP                                                  MAR02780
      ENDIF                                                          MAR02790
*                                                                     MAR02800
11    IF(IEOF .NE. 0) GO TO 9                                        MAR02810
      MOS=MOS1                                                       MAR02820
      LOS=LOS1                                                       MAR02830
      GRADE=GRADE1                                                   MAR02840
      YR=YR1                                                         MAR02850
      INV=INV1                                                       MAR02860
      ATT=ATT1                                                       MAR02870
      IM=IM1                                                         MAR02880
      IL=IL1                                                         MAR02890
      IG=IG1                                                         MAR02900
      IT=IT1                                                         MAR02910
      IF(YR.GT.ST1) FLAG = 1                                         MAR02920
**      CHECK CASE WHERE DATA BEGINS PAST 1ST YR                     MAR02930
      IF(YR .GT. ST1) CINV(IM,IL,IG,IT-1)=.5*FLOAT(INV)              MAR02940
      CALL READER(MOS1,LOS1,GRADE1,YR1,INV1,ATT1, NMOS,SMOS,         MAR02950
     *        NGRD,SGRD, SLOS1,SLOS2, ST1,ST2, IM1,IL1,IG1,IT1,IEOF)  MAR02960
**      ONLY FOR CHECKING PURPOSES. MARK D TO INDICATE SOME YR PRESENT MAR02970
      D(IM,IL,IG)=0                                                  MAR02980
*                                                                     MAR02990
12    IF(.NOT.(IEOF.EQ.0 .AND. MOS1.EQ.MOS .AND. LOS1.EQ.LOS         MAR03000
     *                   .AND. GRADE1.EQ.GRADE)) GO TO 8             MAR03010
**     CENTRAL INV. FOR YR1 - 1                                      MAR03020
       IF(YR1 .GT. YR+1) THEN                                        MAR03030
             CINV(IM1,IL1,IG1,IT1-1)=.5*FLOAT(1NV1)                  MAR03040
             AINV=.5*FLOAT(INV)                                      MAR03050
             I=YR+1                                                  MAR03060
             FLAG=1                                                  MAR03070
       ELSE                                                          MAR03080
             AINV=.5*FLOAT(INV+INV1)                                 MAR03090
       END IF                                                        MAR03100
       CINV(IM,IL,IG,IT)=AMAX1(AINV,ATT)                             MAR03110
       Y(IM,IL,IG,IT)=ATT                                            MAR03120
       YR=YR1                                                        MAR03130
       INV=INV1                                                      MAR03140
       ATT=ATT1                                                      MAR03150
       IM=IM1                                                        MAR03160
       IL=IL1                                                        MAR03170
       IG=IG1                                                        MAR03180
       IT=IT1                                                        MAR03190
       CALL READER(MOS1,LOS1,GRADE1,YR1,INV1,ATT1, NMOS,SMOS,        MAR03200
     *        NGRD,SGRD, SLOS1,SLOS2, ST1,ST2, IM1,IL1,IG1,IT1,IEOF)  MAR03210
       GO TO 12                                                      MAR03220
8     CONTINUE                                                       MAR03230
**      WHEN YEARS MISSING AT THE END                                MAR03240
      IF(YR .LT. LYR) THEN                                           MAR03250
             FLAG=1                                                  MAR03260
             AINV=.5*FLOAT(INV)                                      MAR03270
      ELSE                                                           MAR03280
```

```
            AINV=FLOAT(INV)                                                 MAR03290
         END IF                                                             MAR03300
         CINV(IM,IL,IG,IT)=AMAX1(AINV,ATT)                                  MAR03310
         Y(IM,IL,IG,IT)=ATT                                                 MAR03320
*                                                                           MAR03330
       CC=CINV(IM,IL,IG,IT)                                                 MAR03340
       TEMP=-1.+2.*ATT/(1.+CC)                                              MAR03350
       TEMP1=-1. + 2.*(1.+ATT)/(1.+CC)                                      MAR03360
       CC=ABS(TEMP)                                                         MAR03370
       CC1=ABS(TEMP1)                                                       MAR03380
       IF(CC .GT. 1. .OR. CC1 .GT. 1.) WRITE(6,*) '*** TEMP, TEMP1=',       MAR03390
      *  TEMP,TEMP1,'***MOS,LOS,GR,YR=',SMOS(IM),IL,SGRD(IG),IT             MAR03400
*                                                                           MAR03410
         GO TO 11                                                           MAR03420
9      CONTINUE                                                             MAR03430
*************************************************************************** MAR03440
**       CALCULATE AVERAGE CENTRAL INVENTORY AND MAXIMUM                 ** MAR03450
**                  LIKELIHOOD ESTIMATE.                                 ** MAR03460
*************************************************************************** MAR03470
       DO 10 IM=1,NMOS                                                      MAR03480
        DO 20 IL=1,NLOS                                                     MAR03490
         DO 30 IG=1,NGRD                                                    MAR03500
          SUMY=0                                                            MAR03510
          ACINV(IM,IL,IG)=0                                                 MAR03520
          DO 35 IT=1,T                                                      MAR03530
           SUMY=SUMY+Y(IM,IL,IG,IT)                                         MAR03540
           ACINV(IM,IL,IG)=ACINV(IM,IL,IG)+CINV(IM,IL,IG,IT)                MAR03550
35        CONTINUE                                                          MAR03560
          IF(ACINV(IM,IL,IG) .NE. 0) THEN                                   MAR03570
                              PMLE(IM,IL,IG)=SUMY/ACINV(IM,IL,IG)           MAR03580
          ELSE                                                              MAR03590
                              PMLE(IM,IL,IG)=0                              MAR03600
          ENDIF                                                             MAR03610
          ACINV(IM,IL,IG)=ACINV(IM,IL,IG)/T                                 MAR03620
          DO 40 IT=1,NYR                                                    MAR03630
**        IF ANY CENTRAL INVENTORY>0 (OVER EST. YRS T) THEN D=1             MAR03640
           IF(CINV(IM,IL,IG,IT).GT.0 .AND. IT.LE.T) D(IM,IL,IG)=1           MAR03650
**        IF ANY CENTRAL INVENTORY>0 (OVER VAL. YRS > T) THEN COUNT         MAR03660
           IF(CINV(IM,IL,IG,IT).GT.0 .AND. IT.GT.T) KV(IT-T)=KV(IT-T)+1     MAR03670
**        TRANSFORMATION OF CINV USING THE FREEMAN-TUKEY                    MAR03680
**                DOUBLE ARCSIN FORMULA.                                    MAR03690
          CALL FTT(CINV(IM,IL,IG,IT),Y(IM,IL,IG,IT),X(IM,IL,IG,IT))        MAR03700
          XT(IM,IL,IG,IT)=X(IM,IL,IG,IT) / SQRT(0.5+CINV(IM,IL,IG,IT))     MAR03710
40        CONTINUE                                                          MAR03720
30       CONTINUE                                                           MAR03730
20      CONTINUE                                                            MAR03740
10     CONTINUE                                                             MAR03750
*************************************************************************** MAR03760
**       CHECK FOR MISSING DATA, COMPUTE THE TRANSFORMED                 ** MAR03770
**             CELL AVERAGES AND GRAND MEAN.                             ** MAR03780
*************************************************************************** MAR03790
       XBB=0                                                                MAR03800
       KK=0                                                                 MAR03810
       DO 50 IM=1,NMOS                                                      MAR03820
        DO 60 IL=1,NLOS                                                     MAR03830
         DO 70 IG=1,NGRD                                                    MAR03840
```

```
          IF(D(IM,IL,IG) .EQ. -9999) THEN                                    MAR03850
**        REPORT MISSING COMBINATION IM,IL,IG & CLEAR D ENTRY                MAR03860
                D(IM,IL,IG)=0                                                MAR03870
                ILD=SLOS1+IL-1                                               MAR03880
                FLAG=1                                                       MAR03890
          END IF                                                             MAR03900
          KK=KK+D(IM,IL,IG)                                                  MAR03910
          XB(IM,IL,IG)=0                                                     MAR03920
          XTB(IM,IL,IG)=0                                                    MAR03930
          DO 80 IT=1,T                                                       MAR03940
           XB(IM,IL,IG)=XB(IM,IL,IG)+X(IM,IL,IG,IT)                          MAR03950
           XTB(IM,IL,IG)=XTB(IM,IL,IG)+XT(IM,IL,IG,IT)                       MAR03960
80        CONTINUE                                                           MAR03970
          XB(IM,IL,IG)=XB(IM,IL,IG)/T                                        MAR03980
          XTB(IM,IL,IG)=XTB(IM,IL,IG)/T                                      MAR03990
**      EMPIRICAL BAYES PREPARATION ON THE TRANSFORMED SCALE.                MAR04000
          XEB(IM,IL,IG)=XB(IM,IL,IG)                                         MAR04010
          XTEB(IM,IL,IG)=XTB(IM,IL,IG)                                       MAR04020
*                                                                            MAR04030
          XBB=XBB+D(IM,IL,IG)*XB(IM,IL,IG)                                   MAR04040
70       CONTINUE                                                            MAR04050
60      CONTINUE                                                             MAR04060
50      CONTINUE                                                             MAR04070
        XBB=XBB/KK                                                           MAR04080
*************************************************************************    MAR04090
**        BAYES VARIANCE ITERATION SECTION.                            **    MAR04100
*************************************************************************    MAR04110
        IC=0                                                                 MAR04120
**        COMPUTE TRANSFORMED SCALE CELL VARIANCE.                           MAR04130
499     DO 500 IM=1,NMOS                                                     MAR04140
          DO 510 IL=1,NLOS                                                   MAR04150
            DO 520 IG=1,NGRD                                                 MAR04160
              IF (D(IM,IL,IG).NE.1) GOTO 520                                 MAR04170
              AVSQR = 0.0                                                    MAR04180
              DO 522 IT=1,T                                                  MAR04190
522              AVSQR = AVSQR + SQRT(0.5 + CINV(IM,IL,IG,IT))               MAR04200
              Z = XEB(IM,IL,IG) + (AVSQR/T) * 1.5708                         MAR04210
              V(IM,IL,IG) = VAR(Z)                                           MAR04220
520         CONTINUE                                                         MAR04230
510       CONTINUE                                                           MAR04240
500     CONTINUE                                                             MAR04250
*                                                                            MAR04260
        A=0                                                                  MAR04270
530     AI=A                                                                 MAR04280
        SALPH=0                                                              MAR04290
        DO 531 IM=1,NMOS                                                     MAR04300
          DO 532 IL=1,NLOS                                                   MAR04310
            DO 533 IG=1,NGRD                                                 MAR04320
533           IF (D(IM,IL,IG).EQ.1) SALPH = SALPH + 1./(A + V(IM,IL,IG))     MAR04330
532       CONTINUE                                                           MAR04340
531     CONTINUE                                                             MAR04350
*                                                                            MAR04360
        ZB=0                                                                 MAR04370
**        COMPUTE BAYES ALPHA AND GAMMA PARAMETERS                           MAR04380
        DO 537 IM=1,NMOS                                                     MAR04390
          DO 538 IL=1,NLOS                                                   MAR04400
```

```
          DO 539 IG=1,NGRD                                                    MAR04410
            IF (D(IM,IL,IG).EQ.1) THEN                                        MAR04420
              ALPH = 1./(A + V(IM,IL,IG))                                     MAR04430
              GAM  = ALPH / SALPH                                             MAR04440
              ZB   = ZB + GAM*XB(IM,IL,IG)                                    MAR04450
            ENDIF                                                             MAR04460
539       CONTINUE                                                            MAR04470
538     CONTINUE                                                              MAR04480
537   CONTINUE                                                                MAR04490
          F=0                                                                 MAR04500
          G=0                                                                 MAR04510
      DO 540 IM=1,NMOS                                                        MAR04520
        DO 541 IL=1,NLOS                                                      MAR04530
          DO 542 IG=1,NGRD                                                    MAR04540
            IF (D(IM,IL,IG).EQ.1) THEN                                        MAR04550
              ALPH = 1./(A + V(IM,IL,IG))                                     MAR04560
              F    = F + (ALPH * (XB(IM,IL,IG) - ZB)**2)                      MAR04570
              G    = G + (ALPH * (XB(IM,IL,IG)-ZB))**2                        MAR04580
            ENDIF                                                             MAR04590
542       CONTINUE                                                            MAR04600
541     CONTINUE                                                              MAR04610
540   CONTINUE                                                                MAR04620
**       COMPUTE THE PRIOR VARIANCE (A)                                       MAR04630
      A = A - (KK-1-F)/G                                                      MAR04640
*                                                                             MAR04650
      IF (A.LE.0.) THEN                                                       MAR04660
        A=0                                                                   MAR04670
        GOTO 550                                                              MAR04680
      ENDIF                                                                   MAR04690
      IF (ABS(A-AI).GT..0001) GOTO 530                                        MAR04700
**       COMPUTE THE EMPIRICAL BAYES ATTRITION RATES                          MAR04710
550   DO 570 IM=1,NMOS                                                        MAR04720
        DO 580 IL=1,NLOS                                                      MAR04730
          DO 590 IG=1,NGRD                                                    MAR04740
            IF (D(IM,IL,IG).EQ.1) THEN                                        MAR04750
              DEN = A + V(IM,IL,IG)                                           MAR04760
              XEB(IM,IL,IG) = (A/DEN)*XB(IM,IL,IG)+((DEN-A)/DEN )*ZB          MAR04770
            ENDIF                                                             MAR04780
590       CONTINUE                                                            MAR04790
580     CONTINUE                                                              MAR04800
570   CONTINUE                                                                MAR04810
      IC=IC+1                                                                 MAR04820
      IF (A.EQ.0) GOTO 599                                                    MAR04830
      IF (IC.LT.10) GOTO 499                                                  MAR04840
*                                                                             MAR04850
599   CONTINUE                                                                MAR04860
*                                                                             MAR04870
**       COMPUTE THE EMPIRICAL BAYES SHRINKAGE RATES                          MAR04880
      DO 600 IM=1,NMOS                                                        MAR04890
        DO 601 IL=1,NLOS                                                      MAR04900
          DO 602 IG=1,NGRD                                                    MAR04910
            SHREB(IM,IL,IG) = 0                                               MAR04920
            IF (D(IM,IL,IG).EQ.1) THEN                                        MAR04930
              SHREB(IM,IL,IG) = V(IM,IL,IG)/(A + V(IM,IL,IG))                 MAR04940
            ENDIF                                                             MAR04950
602       CONTINUE                                                            MAR04960
```

```
601      CONTINUE                                                          MAR04970
600    CONTINUE                                                            MAR04980
*************************************************************************  MAR04990
**        ALTERNATE BAYES VARIANCE ITERATION SECTION.                 **  MAR05000
*************************************************************************  MAR05010
**        COMPUTE TRANSFORMED SCALE CELL VARIANCE.                        MAR05020
699    DO 700 IM=1,NMOS                                                    MAR05030
         DO 710 IL=1,NLOS                                                  MAR05040
           DO 720 IG=1,NGRD                                                MAR05050
             VT(IM,IL,IG) = 0.0                                            MAR05060
             IF (D(IM,IL,IG).NE.1) GOTO 720                                MAR05070
             DO 725 IT=1,T                                                 MAR05080
               ZT = X(IM,IL,IG,IT) + 1.5708 * SQRT(0.5 +                   MAR05090
     *              CINV(IM,IL,IG,IT))                                     MAR05100
725            VT(IM,IL,IG) = VT(IM,IL,IG) + VAR(ZT)/(0.5 +                MAR05110
     *                        CINV(IM,IL,IG,IT))                          MAR05120
             VT(IM,IL,IG) = VT(IM,IL,IG)/T**2                              MAR05130
720        CONTINUE                                                        MAR05140
710      CONTINUE                                                          MAR05150
700    CONTINUE                                                            MAR05160
*                                                                          MAR05170
       AT=0                                                               MAR05180
730    ATI=AT                                                             MAR05190
       SALPHT=0                                                           MAR05200
       DO 731 IM=1,NMOS                                                   MAR05210
         DO 732 IL=1,NLOS                                                 MAR05220
           DO 733 IG=1,NGRD                                               MAR05230
733          IF (D(IM,IL,IG).EQ.1) SALPHT= SALPHT+ 1./(AT+ VT(IM,IL,IG)) MAR05240
732      CONTINUE                                                         MAR05250
731    CONTINUE                                                           MAR05260
       ZTB=0                                                              MAR05270
**       COMPUTE BAYES ALPHA AND GAMMA PARAMETERS                        MAR05280
       DO 737 IM=1,NMOS                                                   MAR05290
         DO 738 IL=1,NLOS                                                 MAR05300
           DO 739 IG=1,NGRD                                               MAR05310
             IF (D(IM,IL,IG).EQ.1) THEN                                   MAR05320
               ALPHT= 1./(AT+ VT(IM,IL,IG))                               MAR05330
               GAMT = ALPHT / SALPHT                                      MAR05340
               ZTB  = ZTB+ GAMT*XTB(IM,IL,IG)                             MAR05350
             ENDIF                                                        MAR05360
739        CONTINUE                                                       MAR05370
738      CONTINUE                                                         MAR05380
737    CONTINUE                                                           MAR05390
       PZTB=0.5*(1.+SIN(ZTB))                                             MAR05400
       FT=0                                                               MAR05410
       GT=0                                                               MAR05420
       DO 740 IM=1,NMOS                                                   MAR05430
         DO 741 IL=1,NLOS                                                 MAR05440
           DO 742 IG=1,NGRD                                               MAR05450
             IF (D(IM,IL,IG).EQ.1) THEN                                   MAR05460
               ALPHT= 1./(AT + VT(IM,IL,IG))                              MAR05470
               FT   = FT + (ALPHT * (XTB(IM,IL,IG) - ZTB)**2)             MAR05480
               GT   = GT + (ALPHT * (XTB(IM,IL,IG) - ZTB))**2             MAR05490
             ENDIF                                                        MAR05500
742        CONTINUE                                                       MAR05510
741      CONTINUE                                                         MAR05520
```

```
740   CONTINUE                                                          MAR05530
**       COMPUTE THE PRIOR VARIANCE (A)                                 MAR05540
      AT= AT- (KK-1-FT)/GT                                              MAR05550
*                                                                       MAR05560
      IF (AT.LE.0.) THEN                                                MAR05570
        AT=0                                                            MAR05580
        GOTO 750                                                        MAR05590
      ENDIF                                                             MAR05600
      IF (ABS(AT-ATI).GT..0001) GOTO 730                                MAR05610
**       COMPUTE THE EMPIRICAL BAYES ATTRITION RATES                    MAR05620
750   DO 770 IM=1,NMOS                                                  MAR05630
        DO 780 IL=1,NLOS                                                MAR05640
          DO 790 IG=1,NGRD                                              MAR05650
            IF (D(IM,IL,IG).EQ.1) THEN                                  MAR05660
              DENT = AT + VT(IM,IL,IG)                                  MAR05670
              XTEB(IM,IL,IG) = (AT/DENT)*XTB(IM,IL,IG) +                MAR05680
     *                         ((DENT-AT)/DENT)*ZTB                     MAR05690
            ENDIF                                                       MAR05700
790       CONTINUE                                                      MAR05710
780     CONTINUE                                                        MAR05720
770   CONTINUE                                                          MAR05730
**       COMPUTE THE EMPIRICAL BAYES SHRINKAGE RATES                    MAR05740
      DO 800 IM=1,NMOS                                                  MAR05750
        DO 801 IL=1,NLOS                                                MAR05760
          DO 802 IG=1,NGRD                                              MAR05770
            SHRTEB(IM,IL,IG) = 0                                        MAR05780
            IF (D(IM,IL,IG).EQ.1) THEN                                  MAR05790
              SHRTEB(IM,IL,IG) = VT(IM,IL,IG)/(AT + VT(IM,IL,IG))       MAR05800
            ENDIF                                                       MAR05810
802       CONTINUE                                                      MAR05820
801     CONTINUE                                                        MAR05830
800   CONTINUE                                                          MAR05840
**********************************************************************  MAR05850
**       JAMES-STEIN SHRINKAGE RATES.                               **  MAR05860
**********************************************************************  MAR05870
      SST=0                                                             MAR05880
      SSE=0                                                             MAR05890
      SSB=0                                                             MAR05900
**       COMPUTE THE SUMS OF SQUARES (SST, SSE AND SSB)                 MAR05910
      DO 90 IM=1,NMOS                                                   MAR05920
       DO 95 IL=1,NLOS                                                  MAR05930
        DO 110 IG=1,NGRD                                                MAR05940
         TEMP1=0                                                        MAR05950
         TEMP2=0                                                        MAR05960
         DO 120 IT=1,T                                                  MAR05970
          TEMP1=TEMP1+(X(IM,IL,IG,IT)-XBB)**2                           MAR05980
          TEMP2=TEMP2+(X(IM,IL,IG,IT)-XB(IM,IL,IG))**2                  MAR05990
120      CONTINUE                                                       MAR06000
         SST=SST+D(IM,IL,IG)*TEMP1                                      MAR06010
         SSE=SSE+D(IM,IL,IG)*TEMP2                                      MAR06020
110     CONTINUE                                                        MAR06030
95     CONTINUE                                                         MAR06040
90    CONTINUE                                                          MAR06050
      SSB=SST-SSE                                                       MAR06060
**       COMPUTE THE JAMES-STEIN SHRINKAGE RATES                        MAR06070
      SHRX=(SSE/SSB)* (KK-3)/(KK*(T-1)+2)                               MAR06080
```

```
      SHR=DMIN1(SHRX, 1.D0)                                              MAR06090
      SHR=DMAX1(SHR , 0.D0)                                              MAR06100
      SHR1=T*SHR                                                         MAR06110
*****************************************************************      MAR06120
**       LIMITED TRANSLATION OPTION & JAMES-STEIN                  **    MAR06130
**              ON TRANSFORMED SCALE.                              **    MAR06140
*****************************************************************      MAR06150
      NN=KK*(T-1)                                                        MAR06160
      DO 125 IM=1,NMOS                                                   MAR06170
       DO 130 IL=1,NLOS                                                  MAR06180
        DO 140 IG=1,NGRD                                                 MAR06190
         XJ(IM,IL,IG)=0                                                  MAR06200
         XJ1(IM,IL,IG)=0                                                 MAR06210
         IF (D(IM,IL,IG).EQ.1) THEN                                      MAR06220
           XJ(IM,IL,IG)=XBB+(1-SHR)*(XB(IM,IL,IG)-XBB)                   MAR06230
           XJ1(IM,IL,IG)=XBB+(1-SHR1)*(XB(IM,IL,IG)-XBB)                 MAR06240
C ***      XJLT(IM,IL,IG)=XBB+(1-SHR*RHO(XB(IM,IL,IG),XBB,SSE,           MAR06250
C ***                                SSB,DEE,KK,T,NN))*(XB(IM,IL,IG)-XBB)  MAR06260
           XMLE(IM,IL,IG)=SQRT(.5+ACINV(IM,IL,IG))                       MAR06270
     *                                      *ASIN(2*PMLE(IM,IL,IG)-1)    MAR06280
         ENDIF                                                           MAR06290
140      CONTINUE                                                        MAR06300
130     CONTINUE                                                         MAR06310
125    CONTINUE                                                          MAR06320
*****************************************************************      MAR06330
**       VALIDATION.  RISK EVALUATED IN TRANSFORM SPACE.           **    MAR06340
*****************************************************************      MAR06350
      DO 150 IT=1, NVYR                                                  MAR06360
       RTSCA(IT)=0                                                       MAR06370
       RJ(IT)=0                                                          MAR06380
       RJ1(IT)=0                                                         MAR06390
       RXEB(IT)=0                                                        MAR06400
       RXTEB(IT)=0                                                       MAR06410
C ***  RJLT(IT)=0                                                        MAR06420
       RML(IT)=0                                                         MAR06430
       DO 160 IM=1,NMOS                                                  MAR06440
        DO 170 IL=1,NLOS                                                 MAR06450
         DO 180 IG=1,NGRD                                                MAR06460
          IF(D(IM,IL,IG).EQ.1 .AND. CINV(IM,IL,IG,IT+T).NE.0) THEN       MAR06470
            RTSCA(IT)=RTSCA(IT)+(X(IM,IL,IG,T+IT)-  XB(IM,IL,IG))**2     MAR06480
               RJ(IT)=   RJ(IT)+(X(IM,IL,IG,T+IT)-  XJ(IM,IL,IG))**2     MAR06490
             RJ1(IT)= RJ1(IT)+(X(IM,IL,IG,T+IT)- XJ1(IM,IL,IG))**2      MAR06500
             RXEB(IT)=RXEB(IT)+(X(IM,IL,IG,T+IT)- XEB(IM,IL,IG))**2      MAR06510
             RXTEB(IT)=RXTEB(IT)+((X(IM,IL,IG,T+IT)/SQRT(0.5 +           MAR06520
     *                  CINV(IM,IL,IG,IT+T))- XTEB(IM,IL,IG))**2)       MAR06530
     *                  /VT(IM,IL,IG)                                   MAR06540
C ***       RJLT (IT)= RJLT(IT)+(X(IM,IL,IG,T+IT)-XJLT(IM,IL,IG))**2     MAR06550
            RML  (IT)= RML (IT)+(X(IM,IL,IG,T+IT)-XMLE(IM,IL,IG))**2     MAR06560
          ENDIF                                                          MAR06570
180       CONTINUE                                                       MAR06580
170      CONTINUE                                                        MAR06590
160     CONTINUE                                                         MAR06600
       RTSCA(IT)=RTSCA(IT)/KV(IT)                                        MAR06610
       RJ  (IT)=RJ  (IT)/KV(IT)                                          MAR06620
       RJ1 (IT)=RJ1 (IT)/KV(IT)                                          MAR06630
       RXEB(IT)=RXEB(IT)/KV(IT)                                          MAR06640
```

```
      RXTEB(IT)=RXTEB(IT)/(KV(IT)*T)                                    MAR06650
C ***  RJLT(IT)=RJLT(IT)/KV(IT)                                         MAR06660
      RML (IT)=RML (IT)/KV(IT)                                          MAR06670
      RTJ(IT)=RTSCA(IT)/RJ(IT)                                          MAR06680
C ***  RTJLT(IT)=RTSCA(IT)/RJLT(IT)                                     MAR06690
C ***  RSL(IT)=(RJ(IT)-RTSCA(IT))/(RJLT(IT)-RTSCA(IT))                  MAR06700
150   CONTINUE                                                          MAR06710
*************************************************************************MAR06720
**       DO RISK ANALYSIS ON THE ORIGINAL SCALE.                    **  MAR06730
*************************************************************************MAR06740
      DO 200 IM=1,NMOS                                                  MAR06750
       DO 210 IL=1,NLOS                                                 MAR06760
        DO 220 IG=1,NGRD                                                MAR06770
**       AVERAGE CENTRAL INVENTORY OVER ESTIMATION SET                  MAR06780
         CINVS=0                                                        MAR06790
         DO 230 IT=1,T                                                  MAR06800
         AVCINV=AVCINV + CINV(IM,IL,IG,IT)                              MAR06810
230      CONTINUE                                                       MAR06820
         AVCINV=AVCINV/T                                                MAR06830
         TSCA(IM,IL,IG)=D(IM,IL,IG)*SCINV(  XB(IM,IL,IG), AVCINV)       MAR06840
           PJ(IM,IL,IG)=D(IM,IL,IG)*SCINV(  XJ(IM,IL,IG), AVCINV)       MAR06850
          PJ1(IM,IL,IG)=D(IM,IL,IG)*SCINV( XJ1(IM,IL,IG), AVCINV)       MAR06860
          PEB(IM,IL,IG)=D(IM,IL,IG)*SCINV( XEB(IM,IL,IG), AVCINV)       MAR06870
          PTEB(IM,IL,IG)=D(IM,IL,IG)*0.5*(1. + SIN(XTEB(IM,IL,IG)))     MAR06880
C ***    PJLT(IM,IL,IG)=D(IM,IL,IG)*SCINV(XJLT(IM,IL,IG), AVCINV)       MAR06890
        IF (D(IM,IL,IG).EQ.1) THEN                                      MAR06900
          TSCAM(IM,IL,IG)=MILLER(  XB(IM,IL,IG), AVCINV )               MAR06910
            PJM(IM,IL,IG)=MILLER(  XJ(IM,IL,IG), AVCINV )               MAR06920
           PJ1M(IM,IL,IG)=MILLER( XJ1(IM,IL,IG), AVCINV )               MAR06930
           PEBM(IM,IL,IG)=MILLER( XEB(IM,IL,IG), AVCINV )               MAR06940
              XMIL = XTEB(IM,IL,IG) * SQRT(0.5 + AVCINV)                MAR06950
           PTEBM(IM,IL,IG)=MILLER( XMIL , AVCINV )                      MAR06960
        ENDIF                                                           MAR06970
220     CONTINUE                                                        MAR06980
210    CONTINUE                                                         MAR06990
200   CONTINUE                                                          MAR07000
*************************************************************************MAR07010
**       VALIDATION.  RISK EVALUATED IN ORIGINAL SPACE.             **  MAR07020
*************************************************************************MAR07030
      DO 240 IT=1, NVYR                                                 MAR07040
       RRTSCA(IT)=0                                                     MAR07050
       RRTSAM(IT)=0                                                     MAR07060
       RRJ(IT)=0                                                        MAR07070
       RRJM(IT)=0                                                       MAR07080
       RRPEB(IT)=0                                                      MAR07090
       RRPTEB(IT)=0                                                     MAR07100
       RRPEBM(IT)=0                                                     MAR07110
       RRPTBM(IT)=0                                                     MAR07120
       RRJ1(IT)=0                                                       MAR07130
       RRJ1M(IT)=0                                                      MAR07140
C ***  RRJLT(IT)=0                                                      MAR07150
       RRML(IT)=0                                                       MAR07160
       KP=0                                                             MAR07170
       DO 250 IM=1,NMOS                                                 MAR07180
        DO 260 IL=1,NLOS                                                MAR07190
         DO 270 IG=1,NGRD                                               MAR07200
```

```
            CC=CINV(IM,IL,IG,T+IT)                                              MAR07210
            IF(CC .NE. 0) THEN                                                  MAR07220
                              AR=Y(IM,IL,IG,T+IT)/CC                            MAR07230
            ELSE                                                                MAR07240
                              AR=0                                              MAR07250
            ENDIF                                                               MAR07260
            IF(D(IM,IL,IG) .NE. 0 .AND. PMLE(IM,IL,IG).NE.0                     MAR07270
     *                    .AND. PMLE(IM,IL,IG).NE.1) THEN                      MAR07280
              KP=KP+1                                                           MAR07290
              RRML(IT)=RRML(IT)+(CC*(AR-PMLE(IM,IL,IG))**2)                     MAR07300
     *                          /(PMLE(IM,IL,IG)*(1. - PMLE(IM,IL,IG)))       MAR07310
            ENDIF                                                               MAR07320
     **     PRODUCE CHI-SQUARE MEASURE OF EFFECTIVENESS                         MAR07330
            RRTSCA(IT)= RRTSCA(IT)+CHISQR(D(IM,IL,IG),TSCA(IM,IL,IG),           MAR07340
     *                                    CC,AR)                                MAR07350
            RRTSAM(IT)= RRTSAM(IT)+CHISQR(D(IM,IL,IG),TSCAM(IM,IL,IG),          MAR07360
     *                                    CC,AR)                                MAR07370
            RRJ   (IT)= RRJ   (IT)+CHISQR(D(IM,IL,IG),PJ  (IM,IL,IG),          MAR07380
     *                                    CC,AR)                                MAR07390
            RRJM  (IT)= RRJM  (IT)+CHISQR(D(IM,IL,IG),PJM (IM,IL,IG),          MAR07400
     *                                    CC,AR)                                MAR07410
            RRJ1  (IT)= RRJ1  (IT)+CHISQR(D(IM,IL,IG),PJ1 (IM,IL,IG),          MAR07420
     *                                    CC,AR)                                MAR07430
            RRJ1M (IT)= RRJ1M (IT)+CHISQR(D(IM,IL,IG),PJ1M(IM,IL,IG),          MAR07440
     *                                    CC,AR)                                MAR07450
            RRPEB (IT)= RRPEB (IT)+CHISQR(D(IM,IL,IG),PEB (IM,IL,IG),          MAR07460
     *                                    CC,AR)                                MAR07470
            RRPTEB (IT)= RRPTEB (IT)+CHISQR(D(IM,IL,IG),PTEB (IM,IL,IG),       MAR07480
     *                                    CC,AR)                                MAR07490
            RRPEBM(IT)= RRPEBM(IT)+CHISQR(D(IM,IL,IG),PEBM(IM,IL,IG),          MAR07500
     *                                    CC,AR)                                MAR07510
            RRPTBM(IT)= RRPTBM(IT)+CHISQR(D(IM,IL,IG),PTEBM(IM,IL,IG),         MAR07520
     *                                    CC,AR)                                MAR07530
     C ***  RRJLT (IT)= RRJLT (IT)+CHISQR(D(IM,IL,IG),PJLT(IM,IL,IG),          MAR07540
     C   *                                 CC,AR)                                MAR07550
     **     PRODUCE MEAN ABSOLUTE DEVIATION MEASURE OF EFFECTIVENESS            MAR07560
            IF(D(IM,IL,IG) .NE. 0 .AND. PMLE(IM,IL,IG).NE.0                     MAR07570
     *                    .AND. PMLE(IM,IL,IG).NE.1) THEN                      MAR07580
              CALL UOM( CC, AR, PMLE(IM,IL,IG), MOPMLE(IT),                     MAR07590
     *                  MUPMLE(IT), KVOPML(IT), KVUPML(IT), KVDPML(IT))       MAR07600
            ENDIF                                                               MAR07610
                                                                                MAR07620
            IF(CC.NE.0.) THEN                                                   MAR07630
              CALL UOM( CC, AR, TSCA(IM,IL,IG), MOTSCA(IT),                     MAR07640
     *                  MUTSCA(IT), KVOTSA(IT), KVUTSA(IT), KVDTSA(IT))       MAR07650
                                                                                MAR07660
              CALL UOM( CC, AR, TSCAM(IM,IL,IG), MOTSAM(IT),                    MAR07670
     *                  MUTSAM(IT), KVOTSM(IT), KVUTSM(IT), KVDTSM(IT))       MAR07680
                                                                                MAR07690
              CALL UOM( CC, AR, PJ(IM,IL,IG), MOPJ(IT),                        MAR07700
     *                  MUPJ(IT), KVOPJ(IT), KVUPJ(IT), KVDPJ(IT))            MAR07710
                                                                                MAR07720
              CALL UOM( CC, AR, PJM(IM,IL,IG), MOPJM(IT),                       MAR07730
     *                  MUPJM(IT), KVOPJM(IT), KVUPJM(IT), KVDPJM(IT))        MAR07740
                                                                                MAR07750
              CALL UOM( CC, AR, PJ1(IM,IL,IG), MOPJ1(IT),                       MAR07760
```

```
     *                MUPJ1(IT), KVOPJ1(IT), KVUPJ1(IT), KVDPJ1(IT))    MAR07770
                                                                         MAR07780
           CALL UOM( CC, AR, PJ1M(IM,IL,IG), MOPJ1M(IT),                MAR07790
     *                MUPJ1M(IT), KVOP1M(IT), KVUP1M(IT), KVDP1M(IT))   MAR07800
                                                                         MAR07810
           CALL UOM( CC, AR, PEB(IM,IL,IG), MOPEB(IT),                  MAR07820
     *                MUPEB(IT), KVOPEB(IT), KVUPEB(IT), KVDPEB(IT))    MAR07830
                                                                         MAR07840
           CALL UOM( CC, AR, PTEB(IM,IL,IG), MOPTEB(IT),                MAR07850
     *                MUPTEB(IT), KVOPTB(IT), KVUPTB(IT), KVDPTB(IT))   MAR07860
                                                                         MAR07870
           CALL UOM( CC, AR, PEBM(IM,IL,IG), MOPEBM(IT),                MAR07880
     *                MUPEBM(IT), KVOPBM(IT), KVUPBM(IT), KVDPBM(IT))   MAR07890
                                                                         MAR07900
           CALL UOM( CC, AR, PTEBM(IM,IL,IG), MOPTBM(IT),               MAR07910
     *                MUPTBM(IT), KVOBTM(IT), KVUBTM(IT), KVDBTM(IT))   MAR07920
         ENDIF                                                           MAR07930
                                                                         MAR07940
270       CONTINUE                                                      MAR07950
260      CONTINUE                                                       MAR07960
250     CONTINUE                                                        MAR07970
        IF (KP .EQ. 0) THEN                                             MAR07980
           FF = 0                                                       MAR07990
        ELSE                                                            MAR08000
           FF=FLOAT(KV(IT))/FLOAT(KP)                                   MAR08010
        ENDIF                                                           MAR08020
        RRML(IT)=RRML(IT)*FF                                            MAR08030
240    CONTINUE                                                         MAR08040
**        COMPUTE TIME AVERAGE MAD                                      MAR08050
**        MEASURE OF EFFECTIVENESS                                      MAR08060
       DO 300 IT=1, NVYR                                                MAR08070
         MUPMLE(IT) = MUPMLE(IT)/KV(IT)                                 MAR08080
         MUTSCA(IT) = MUTSCA(IT)/KV(IT)                                 MAR08090
         MUTSAM(IT) = MUTSAM(IT)/KV(IT)                                 MAR08100
         MUPJ  (IT) = MUPJ (IT)/KV(IT)                                  MAR08110
         MUPJM (IT) = MUPJM(IT)/KV(IT)                                  MAR08120
         MUPJ1 (IT) = MUPJ1(IT)/KV(IT)                                  MAR08130
         MUPJ1M(IT) = MUPJ1M(IT)/KV(IT)                                 MAR08140
         MUPEB (IT) = MUPEB(IT)/KV(IT)                                  MAR08150
         MUPTEB (IT) = MUPTEB(IT)/KV(IT)                                MAR08160
         MUPEBM(IT) = MUPEBM(IT)/KV(IT)                                 MAR08170
         MUPTBM(IT) = MUPTBM(IT)/KV(IT)                                 MAR08180
         MOPMLE(IT) = MOPMLE(IT)/KV(IT)                                 MAR08190
         MOTSCA(IT) = MOTSCA(IT)/KV(IT)                                 MAR08200
         MOTSAM(IT) = MOTSAM(IT)/KV(IT)                                 MAR08210
         MOPJ  (IT) = MOPJ (IT)/KV(IT)                                  MAR08220
         MOPJM (IT) = MOPJM(IT)/KV(IT)                                  MAR08230
         MOPJ1 (IT) = MOPJ1(IT)/KV(IT)                                  MAR08240
         MOPJ1M(IT) = MOPJ1M(IT)/KV(IT)                                 MAR08250
         MOPEB (IT) = MOPEB(IT)/KV(IT)                                  MAR08260
         MOPTEB (IT) = MOPTEB(IT)/KV(IT)                                MAR08270
         MOPEBM(IT) = MOPEBM(IT)/KV(IT)                                 MAR08280
         MOPTBM(IT) = MOPTBM(IT)/KV(IT)                                 MAR08290
         MDPMLE(IT) = MUPMLE(IT) + MOPMLE(IT)                           MAR08300
         MDTSCA(IT) = MUTSCA(IT) + MOTSCA(IT)                           MAR08310
         MDTSAM(IT) = MUTSAM(IT) + MOTSAM(IT)                           MAR08320
```

```
        MDPJ  (IT) = MUPJ  (IT) + MOPJ  (IT)                              MAR08330
        MDPJM (IT) = MUPJM (IT) + MOPJM (IT)                              MAR08340
        MDPJ1 (IT) = MUPJ1 (IT) + MOPJ1 (IT)                              MAR08350
        MDPJ1M(IT) = MUPJ1M(IT) + MOPJ1M(IT)                              MAR08360
        MDPEB (IT) = MUPEB (IT) + MOPEB (IT)                              MAR08370
        MDPTEB (IT) = MUPTEB (IT) + MOPTEB (IT)                           MAR08380
        MDPEBM(IT) = MUPEBM(IT) + MOPEBM(IT)                              MAR08390
        MDPTBM(IT) = MUPTBM(IT) + MOPTBM(IT)                              MAR08400
300   CONTINUE                                                            MAR08410
**********************************************************************    MAR08420
**       PRINT SECTION FOR ALL OUTPUT.                              **    MAR08430
**********************************************************************    MAR08440
**       PRINT GROUP CHARACTERISTICS                                       MAR08450
      WRITE(6,'(1X,''TEN OR LESS BAYES VARIANCE ITERATIONS'')')           MAR08460
      WRITE(6,'(1X,I2,'' BAYES VARIANCE ITERATION(S)'')') IC              MAR08470
      WRITE(6,'(1X,''XBB '',F12.6)') XBB                                  MAR08480
      WRITE(6,905) T,KK,SSB,SSE,SHRX,SHR,ZB,ZTB,PZTB,A,AT                 MAR08490
905   FORMAT(' GROUP CHARACTERISTICS', /,' T=',I6,  /,' K=',I6,           MAR08500
     *   /,'  SSB=',F12.4, /,'  SSE=',F12.4, /,' SHRX=',F12.4,           MAR08510
     *   /,'  SHR=',F12.4, /,'   ZB=',F12.4, /,'  ZTB=',F12.4,           MAR08520
     *   /,' PZTB=',F12.4, /,'    A=',F12.4, /,'   AT=',F12.4)           MAR08530
**       PRINT RESULTS FOR TRANSFORM SPACE                                 MAR08540
      WRITE(6,*) ' RESULTS FOR TRANSFORM SPACE'                           MAR08550
      WRITE(6,906) (IVYR(I), I=1, NVYR)                                   MAR08560
906   FORMAT(14X,10I12)                                                   MAR08570
      WRITE(6,907) 'RTSCA ', (RTSCA(IT), IT=1,NVYR)                       MAR08580
907   FORMAT(' ',A14,10F12.4)                                             MAR08590
      WRITE(6,907) 'RJ    ', (RJ   (IT), IT=1,NVYR)                       MAR08600
      WRITE(6,907) 'RJ1   ', (RJ1  (IT), IT=1,NVYR)                       MAR08610
      WRITE(6,907) 'RXEB  ', (RXEB (IT), IT=1,NVYR)                       MAR08620
      WRITE(6,907) 'RXTEB ', (RXTEB (IT), IT=1,NVYR)                      MAR08630
C *** WRITE(6,907) 'RJLT  ', (RJLT (IT), IT=1,NVYR)                       MAR08640
      WRITE(6,907) 'RML   ', (RML  (IT), IT=1,NVYR)                       MAR08650
C *** WRITE(6,907) 'RSL   ', (RSL  (IT), IT=1,NVYR)                       MAR08660
C *** WRITE(6,907) 'RTJ   ', (RTJ  (IT), IT=1,NVYR)                       MAR08670
C *** WRITE(6,907) 'RTJLT ', (RTJLT(IT), IT=1,NVYR)                       MAR08680
**       PRINT RESULTS FOR ORIGINAL SPACE                                  MAR08690
      WRITE(6,*)    ' '                                                   MAR08700
      WRITE(6,*) ' RESULTS FOR ORIGINAL SPACE'                            MAR08710
      WRITE(6,907) 'RRML  ', (RRML (IT), IT=1,NVYR)                       MAR08720
      WRITE(6,907) 'RRTSCA', (RRTSCA(IT), IT=1,NVYR)                      MAR08730
      WRITE(6,907) 'RRTSAM', (RRTSAM(IT), IT=1,NVYR)                      MAR08740
      WRITE(6,907) 'RRJ   ', (RRJ  (IT), IT=1,NVYR)                       MAR08750
      WRITE(6,907) 'RRJM  ', (RRJM (IT), IT=1,NVYR)                       MAR08760
      WRITE(6,907) 'RRJ1  ', (RRJ1 (IT), IT=1,NVYR)                       MAR08770
      WRITE(6,907) 'RRJ1M ', (RRJ1M (IT), IT=1,NVYR)                      MAR08780
      WRITE(6,907) 'RRPEB ', (RRPEB(IT), IT=1,NVYR)                       MAR08790
      WRITE(6,907) 'RRPTEB ', (RRPTEB(IT), IT=1,NVYR)                     MAR08800
      WRITE(6,907) 'RRPEBM', (RRPEBM(IT), IT=1,NVYR)                      MAR08810
      WRITE(6,907) 'RRPTBM', (RRPTBM(IT), IT=1,NVYR)                      MAR08820
C *** WRITE(6,907) 'RRJLT ', (RRJLT(IT), IT=1,NVYR)                       MAR08830
      WRITE(6,908) 'KV    ', (KV   (IT), IT=1,NVYR)                       MAR08840
908   FORMAT(' ',A14,10I12)                                               MAR08850
      WRITE(6,'(9X,''KP    '',I12)') KP                                   MAR08860
      WRITE(6,*)    ' '                                                   MAR08870
**       PRINT MAD CALCULATIONS                                            MAR08880
```

```
      WRITE(6,'(2X,''PMLE'')')                                          MAR08890
      CALL WRMAD(IVYR,NVYR,MUPMLE,KVUPML,MOPMLE,KVOPML,MDPMLE,KVDPML)   MAR08900
      WRITE(6,'(2X,''TSCA'')')                                          MAR08910
      CALL WRMAD(IVYR,NVYR,MUTSCA,KVUTSA,MOTSCA,KVOTSA,MDTSCA,KVDTSA)   MAR08920
      WRITE(6,'(2X,''TSCAM'')')                                         MAR08930
      CALL WRMAD(IVYR,NVYR,MUTSAM,KVUTSM,MOTSAM,KVOTSM,MDTSAM,KVDTSM)   MAR08940
      WRITE(6,'(2X,''PJ'')')                                            MAR08950
      CALL WRMAD(IVYR,NVYR,MUPJ,KVUPJ,MOPJ,KVOPJ,MDPJ,KVDPJ)            MAR08960
      WRITE(6,'(2X,''PJM'')')                                           MAR08970
      CALL WRMAD(IVYR,NVYR,MUPJM,KVUPJM,MOPJM,KVOPJM,MDPJM,KVDPJM)      MAR08980
      WRITE(6,'(2X,''PJ1'')')                                           MAR08990
      CALL WRMAD(IVYR,NVYR,MUPJ1,KVUPJ1,MOPJ1,KVOPJ1,MDPJ1,KVDPJ1)      MAR09000
      WRITE(6,'(2X,''PJ1M'')')                                          MAR09010
      CALL WRMAD(IVYR,NVYR,MUPJ1M,KVUP1M,MOPJ1M,KVOP1M,MDPJ1M,KVDP1M)   MAR09020
      WRITE(6,'(2X,''PEB'')')                                           MAR09030
      CALL WRMAD(IVYR,NVYR,MUPEB,KVUPEB,MOPEB,KVOPEB,MDPEB,KVDPEB)      MAR09040
      WRITE(6,'(2X,''PEBM'')')                                          MAR09050
      CALL WRMAD(IVYR,NVYR,MUPEBM,KVUPBM,MOPEBM,KVOPBM,MDPEBM,KVDPBM)   MAR09060
      WRITE(6,'(2X,''PTEB'')')                                          MAR09070
      CALL WRMAD(IVYR,NVYR,MUPTEB,KVUPTB,MOPTEB,KVOPTB,MDPTEB,KVDPTB)   MAR09080
      WRITE(6,'(2X,''PTEBM'')')                                         MAR09090
      CALL WRMAD(IVYR,NVYR,MUPTBM,KVUBTM,MOPTBM,KVOBTM,MDPTBM,KVDBTM)   MAR09100
      WRITE(6,*)  ' '                                                   MAR09110
**       PRINT THE CENTRAL INVENTORY MATRIX                             MAR09120
      WRITE(6,*) 'CINV MATRIX'                                          MAR09130
      DO 1006 IT=1,NYR                                                  MAR09140
       WRITE(6,*) 'YEAR',IT                                             MAR09150
       DO 1005 IL=1,NLOS                                                MAR09160
        WRITE(6,1007)(CINV(IM,IL,1,IT),IM=1,NMOS)                       MAR09170
1005   CONTINUE                                                         MAR09180
1006  CONTINUE                                                          MAR09190
1007  FORMAT(15(1X,F7.2))                                               MAR09200
      WRITE(6,*)  ' '                                                   MAR09210
**       PRINT THE ATTRITION MATRIX                                     MAR09220
      WRITE(6,*) 'Y MATRIX'                                             MAR09230
      DO 1016 IT=1,NYR                                                  MAR09240
       WRITE(6,*) 'YEAR',IT                                             MAR09250
       DO 1015 IL=1,NLOS                                                MAR09260
        WRITE(6,1007)(Y(IM,IL,1,IT),IM=1,NMOS)                          MAR09270
1015   CONTINUE                                                         MAR09280
1016  CONTINUE                                                          MAR09290
      WRITE(6,*)  ' '                                                   MAR09300
**       PRINT ALL ARRAYS                                               MAR09310
      WRITE(6,'(2X,''ARRAY TSCA'')')                                    MAR09320
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,TSCA)                 MAR09330
      WRITE(6,'(2X,''ARRAY TSCAM'')')                                   MAR09340
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,TSCAM)                MAR09350
      WRITE(6,'(2X,''ARRAY SHREB'')')                                   MAR09360
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,SHREB)                MAR09370
      WRITE(6,'(2X,''ARRAY SHRTEB'')')                                  MAR09380
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,SHRTEB)               MAR09390
      WRITE(6,'(2X,''ARRAY PMLE'')')                                    MAR09400
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PMLE)                 MAR09410
      WRITE(6,'(2X,''ARRAY PJ'')')                                      MAR09420
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PJ)                   MAR09430
      WRITE(6,'(2X,''ARRAY PJM'')')                                     MAR09440
```

```
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PJM)                  MAR09450
      WRITE(6,'(2X,''ARRAY PJ1'')')                                     MAR09460
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PJ1)                  MAR09470
      WRITE(6,'(2X,''ARRAY PJ1M'')')                                    MAR09480
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PJ1M)                 MAR09490
      WRITE(6,'(2X,''ARRAY PEB'')')                                     MAR09500
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PEB)                  MAR09510
      WRITE(6,'(2X,''ARRAY PTEB'')')                                    MAR09520
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PTEB)                 MAR09530
      WRITE(6,'(2X,''ARRAY PEBM'')')                                    MAR09540
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PEBM)                 MAR09550
      WRITE(6,'(2X,''ARRAY PTEBM'')')                                   MAR09560
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,PTEBM)                MAR09570
      WRITE(6,'(2X,''ARRAY V'')')                                       MAR09580
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,V)                    MAR09590
      WRITE(6,'(2X,''ARRAY XB'')')                                      MAR09600
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,XB)                   MAR09610
      WRITE(6,'(2X,''ARRAY VT'')')                                      MAR09620
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,VT)                   MAR09630
      WRITE(6,'(2X,''ARRAY XTB'')')                                     MAR09640
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,XTB)                  MAR09650
      WRITE(6,'(2X,''ARRAY XJ'')')                                      MAR09660
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,XJ)                   MAR09670
      WRITE(6,'(2X,''ARRAY XJ1'')')                                     MAR09680
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,XJ1)                  MAR09690
      WRITE(6,'(2X,''ARRAY XEB'')')                                     MAR09700
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,XEB)                  MAR09710
      WRITE(6,'(2X,''ARRAY XTEB'')')                                    MAR09720
      CALL ARRAY(MXMOS,MXLOS,MXGRD,NMOS,NLOS,NGRD,XTEB)                 MAR09730
*                                                                       MAR09740
      END                                                               MAR09750
*****************************************************************       MAR09760
      SUBROUTINE WRMAD(IYR,NYR,MU,KU,MO,KO,MD,KD)                       MAR09770
*****************************************************************       MAR09780
      INTEGER IYR(NYR),KU(NYR),KO(NYR),KD(NYR)                          MAR09790
      REAL    MU(NYR),MO(NYR),MD(NYR)                                   MAR09800
      WRITE(6,'(/9X,''DEVIATION'',T26,I2,T42,I2,T59,I2/)')              MAR09810
     *        (IYR(IW),IW=1,NYR)                                         MAR09820
      WRITE(6,'(9X,''UNDERAGE'',T22,F9.6,'' ('',I2,'')'',T38,F9.6,      MAR09830
     *        '' ('',I2,'')'',T55,F9.6,'' ('',I2,'')'')')               MAR09840
     *        MU(1),KU(1),MU(2),KU(2),MU(3),KU(3)                       MAR09850
      WRITE(6,'(9X,''OVERAGE'',T22,F9.6,'' ('',I2,'')'',T38,F9.6,       MAR09860
     *        '' ('',I2,'')'',T55,F9.6,'' ('',I2,'')'')')               MAR09870
     *        MO(1),KO(1),MO(2),KO(2),MO(3),KO(3)                       MAR09880
      WRITE(6,'(9X,''TOTAL'',T22,F9.6,'' ('',I2,'')'',T38,F9.6,         MAR09890
     *        '' ('',I2,'')'',T55,F9.6,'' ('',I2,'')'')')               MAR09900
     *        MD(1),KD(1),MD(2),KD(2),MD(3),KD(3)                       MAR09910
      WRITE(6,'(/)')                                                    MAR09920
      RETURN                                                            MAR09930
      END                                                               MAR09940
*****************************************************************       MAR09950
      SUBROUTINE ARRAY(MM,ML,MG,NM,NL,NG,OBJ1)                          MAR09960
*****************************************************************       MAR09970
      REAL OBJ1(MM,ML,MG)                                               MAR09980
      DO 10 IG1=1,NG                                                    MAR09990
```

```
        DO 10 IL1=1,NL                                                  MAR10000
10         WRITE(6,'(2X,15(F7.3,1X))') (OBJ1(IM1,IL1,IG1),IM1=1,NM)     MAR10010
      WRITE(6,'(/)')                                                    MAR10020
      RETURN                                                            MAR10030
      END                                                               MAR10040
*********************************************************************** MAR10050
      SUBROUTINE FTT(CINV,Y, X)                                         MAR10060
*********************************************************************** MAR10070
      REAL CINV,X,Y                                                     MAR10080
      TEMP=Y/(1+CINV)                                                   MAR10090
      TEMP1=(1.+Y)/(1+CINV)                                             MAR10100
      TEMP=SQRT(.5+CINV)*.5*(ASIN(-1.+2.*TEMP) +                        MAR10110
     *                       ASIN(-1.+2.*TEMP1))                        MAR10120
      X=TEMP                                                            MAR10130
      END                                                               MAR10140
*********************************************************************** MAR10150
      SUBROUTINE UOM(VINV,ACT,OBJ,OVER,UNDER,KO,KU,KD)                  MAR10160
*********************************************************************** MAR10170
      REAL OBJ                                                          MAR10180
      IF (ACT .GT. OBJ) THEN                                            MAR10190
        UNDER = UNDER+VINV*(ACT-OBJ)                                    MAR10200
        KU=KU+1                                                         MAR10210
      ELSE                                                              MAR10220
        OVER  = OVER +VINV*(OBJ-ACT)                                    MAR10230
        KO=KO+1                                                         MAR10240
      ENDIF                                                             MAR10250
      KD=KD+1                                                           MAR10260
      RETURN                                                            MAR10270
      END                                                               MAR10280
*********************************************************************** MAR10290
      REAL FUNCTION RHO(XB,XBB,SSE,SSB,DEE,KK,T,NN)                     MAR10300
*********************************************************************** MAR10310
      INTEGER T                                                         MAR10320
      REAL*8 SIGSQ, V, U                                                MAR10330
      SIGSQ=SSE/(NN+2)                                                  MAR10340
      V=SSB/T                                                           MAR10350
      U=KK*(KK-3)*(XB-XBB)**2/(V*(KK-1))                                MAR10360
      RHO=DMIN1(1.D0, DEE/DSQRT(U))                                     MAR10370
      END                                                               MAR10380
*********************************************************************** MAR10390
      REAL FUNCTION MILLER( X, AVGINV )                                 MAR10400
*********************************************************************** MAR10410
      TM = X/SQRT(AVGINV+.5)+1.570796                                   MAR10420
      TEMP1 = ASIN(SQRT(1./(AVGINV+1)))                                 MAR10430
      IF (TM .LT. TEMP1) THEN                                           MAR10440
        MILLER = 0.0                                                    MAR10450
        GOTO 1                                                          MAR10460
      ENDIF                                                             MAR10470
      IF (TM .GT. 3.14159-TEMP1) THEN                                   MAR10480
        MILLER = 1.0                                                    MAR10490
        GOTO 1                                                          MAR10500
      ENDIF                                                             MAR10510
      IF (COS(TM) .EQ. 0.0) THEN                                        MAR10520
        MILLER = 0.5                                                    MAR10530
        GOTO 1                                                          MAR10540
      ENDIF                                                             MAR10550
```

```
      IF (COS(TM) .GT. 0.0) SIGNUM = 1.                                  MAR10560
      IF (COS(TM) .LT. 0.0) SIGNUM = -1.                                 MAR10570
      TEMP2=1-(SIN(TM)+(SIN(TM)-(1./SIN(TM)))/AVGINV)**2                 MAR10580
      IF (TEMP2.LT.0.0) TEMP2=0.0                                        MAR10590
      MILLER=.5*(1.-SIGNUM*TEMP2**.5)                                    MAR10600
1     END                                                                MAR10610
*************************************************************************MAR10620
      REAL FUNCTION SCINV(X,A)                                           MAR10630
*************************************************************************MAR10640
**       SCALE INVERSION (TRANSFORM ==> ORIGINAL SCALE)                  MAR10650
      R=X/SQRT(A+.5)                                                     MAR10660
      IF(R .LT. -1.570796) THEN                                          MAR10670
          SCINV=0.                                                       MAR10680
      ELSE IF(R .GT.  1.570796) THEN                                     MAR10690
          SCINV=1.                                                       MAR10700
      ELSE                                                               MAR10710
          SCINV=.5*(1.+SIN(R))                                           MAR10720
      ENDIF                                                              MAR10730
      END                                                                MAR10740
*************************** ** ***************************************** MAR10750
      REAL FUNCTION VAR( ZZ )                                            MAR10760
*************************************************************************MAR10770
      DATA AA/1.6835/, B1/-.8934/, B2/.8991/                             MAR10780
      IF ( ZZ .GT. 2.2) THEN                                             MAR10790
        VAR = 1.                                                         MAR10800
      ELSE                                                               MAR10810
        IF ( ZZ .LT. 1.001) THEN                                         MAR10820
          VAR = .05                                                      MAR10830
        ELSE                                                             MAR10840
          VAR = AA*(ZZ**B1)*(ZZ-1.)**B2                                  MAR10850
        ENDIF                                                            MAR10860
      ENDIF                                                              MAR10870
      END                                                                MAR10880
*************************************************************************MAR10890
      REAL FUNCTION CHISQR(ID,OBJ,INV,ACT)                               MAR10900
*************************************************************************MAR10910
      REAL OBJ,INV                                                       MAR10920
      IF (ID.NE.0 .AND. OBJ.NE.0. .AND. OBJ.NE.1.)                       MAR10930
     *  CHISQR = (INV*(ACT-OBJ)**2)/(OBJ*(1.-OBJ))                       MAR10940
      END                                                                MAR10950
*************************************************************************MAR10960
      SUBROUTINE READER(MOS,LOS,GRADE,YR,INV,Y, NMOS,SMOS,               MAR10970
     *   NGRD,SGRD, SLOS1,SLOS2, ST1,ST2, IM,IL,IG,IT, IEOF)             MAR10980
*************************************************************************MAR10990
**       READ RECORD AND ACCUMULATE LOSSES                               MAR11000
      INTEGER MOS, LOS, GRADE, YR, INV,  LOSS(8)                         MAR11010
      REAL Y                                                             MAR11020
      INTEGER ST1, ST2                                                   MAR11030
      INTEGER SLOS1, SLOS2                                               MAR11040
      INTEGER SMOS(40), NMOS                                             MAR11050
      INTEGER SGRD(10), NGRD                                             MAR11060
      ICNT=0                                                             MAR11070
1     READ(1,100,END=999) YR,MOS,GRADE,LOS,INV,(LOSS(I),I=1,8)           MAR11080
100   FORMAT(4I2,9I5)                                                    MAR11090
**       CHECK IF RECORD MEETS SELECTION CRITERIA.  OTHERWISE REJECT.    MAR11100
      IM=0                                                               MAR11110
```

```
      DO 10 I=1,NMOS                                                    MAR11120
       IF(MOS .EQ. SMOS(I)) THEN                                        MAR11130
              IM=I                                                      MAR11140
              GO TO 20                                                  MAR11150
       END IF                                                           MAR11160
10    CONTINUE                                                         MAR11170
      GO TO 1                                                          MAR11180
*                                                                      MAR11190
20    CONTINUE                                                         MAR11200
      DO 30 I=1,NGRD                                                   MAR11210
       IF(GRADE .EQ. SGRD(I)) THEN                                     MAR11220
              IG=I                                                      MAR11230
              GO TO 40                                                  MAR11240
       END IF                                                          MAR11250
30    CONTINUE                                                         MAR11260
      GO TO 1                                                          MAR11270
40    CONTINUE                                                         MAR11280
*                                                                      MAR11290
      IF(LOS .LT. SLOS1 .OR. LOS .GT. SLOS2) GO TO 1                   MAR11300
      IL=LOS-SLOS1+1                                                   MAR11310
**       YEARS OVER ST2 ARE USED FOR VALIDATION                        MAR11320
      IF(YR .LT. ST1) GO TO 1                                          MAR11330
      IT=YR-ST1+1                                                      MAR11340
**       COMPUTE TOTAL LOSS                                            MAR11350
      Y=0                                                              MAR11360
      DO 50 I=1,8                                                      MAR11370
       Y=Y + LOSS(I)                                                   MAR11380
50    CONTINUE                                                         MAR11390
*                                                                      MAR11400
      RETURN                                                           MAR11410
999   IEOF=1                                                           MAR11420
      END                                                              MAR11430
```

APPENDIX C. SIMULATION FOR CHOICE OF AVERAGE INVENTORY VALUES

```
      PROGRAM MCSIM                                                          MCS00010
*************************************************************************** MCS00020
*     SIMULATION COMPARISON OF INVENTORY MEAN PERFORMANCE               *   MCS00030
*                                                                       *   MCS00040
*     PURPOSE:  TO IDENTIFY THE BEST MEAN VALUE (ARITHMETIC,            *   MCS00050
*               GEOMETRIC AND HARMONIC) FOR USE IN THE INVERSION OF     *   MCS00060
*               TRANSFORMED DATA TO THE ORIGINAL SPACE.                 *   MCS00070
*     PROGRAMMER: CAPT C R DICKINSON, USMC                              *   MCS00080
*     VARIABLE DESCRIPTION:                                             *   MCS00090
*               LAMBDA - PARAMETER USED IN THE POISSON RANDOM NUMBER    *   MCS00100
*                        GENERATOR                                      *   MCS00110
*               TRIALS - VECTOR OF INVENTORY VALUES GENERATED FROM      *   MCS00120
*                        THE POISSON RANDOM NUMBER GENERATOR            *   MCS00130
*               REPS   - NUMBER OF INVENTORY VALUES TO BE GENERATED     *   MCS00140
*               BASIC  - BASIC INVERSION OF TRANSFORMED VALUES          *   MCS00150
*               FTE1   - FREEMAN-TUKEY EXACT INVERSION WITH             *   MCS00160
*                                                                       *   MCS00170
*               FTE2   - FREEMAN-TUKEY EXACT INVERSION WITH             *   MCS00180
*                                                                       *   MCS00190
*               Z      - TRANSFORMED SCALE VARIABLE                     *   MCS00200
*               MEANS  - VECTOR OF ARITHMETIC, GEOMETRIC AND HARMONIC   *   MCS00210
*                        MEANS                                          *   MCS00220
*               TRSUM  - SUMMATION OF TRIALS FOR USE IN THE             *   MCS00230
*                        ARITHMETIC MEAN                                *   MCS00240
*               TRMULT - MULTIPLICATION OF TRIALS FOR USE IN THE        *   MCS00250
*                        GEOMETRIC MEAN                                 *   MCS00260
*               TRINV  - SUMMATION OF THE INVERSE OF TRIALS FOR USE     *   MCS00270
*                        IN THE HARMONIC MEAN                           *   MCS00280
*     INPUT/OUTPUT:                                                     *   MCS00290
*               INPUT - HARD CODED PARAMETERS                           *   MCS00300
*              OUTPUT - SIM OUT A1 (DISK FILE 03)                       *   MCS00310
*************************************************************************** MCS00320
*                                                                           MCS00330
***      VARIABLE DECLARATION                                               MCS00340
      INTEGER TRIALS(100),Y(100),REPS(3),PROB,R,                            MCS00350
     C     COUNT,TRSUM,YTEMP,YSUM                                           MCS00360
      REAL Z(100),T(100),MEAN(3),BASIC(3),DIFF(3),LAMBDA(5),                MCS00370
     C     P(4),FTE1(3),FTE2(3),TRMULT                                      MCS00380
      DOUBLE PRECISION DPSEED                                               MCS00390
*                                                                           MCS00400
      CALL EXCMS('FILEDEF 02 DISK REPLIC LISTING A1')                       MCS00410
*                                                                           MCS00420
***      VARIABLE INITIALIZATION                                            MCS00430
      PI=3.14159                                                            MCS00440
      DPSEED=889246.D0                                                      MCS00450
      LAM=5                                                                 MCS00460
      R=3                                                                   MCS00470
      PROB=4                                                                MCS00480
      COUNT=2                                                               MCS00490
```

```
      DATA REPS/20,40,60/                                              MCS00500
      DATA LAMBDA/1.,2.,4.,8.,16./                                     MCS00510
      DATA P/.05,.1,.2,.4/                                             MCS00520
*                                                                      MCS00530
***       HEADER PRINT                                                 MCS00540
      WRITE (02,'(/T6,''LAMBDA'',T15,''REPS'',T27,''P'',               MCS00550
     C      T34,''METHOD'',T46,''ARITHMETIC'',T59,                     MCS00560
     C      ''GEOMETRIC'',T71,''HARMONIC''/)')                         MCS00570
*                                                                      MCS00580
***       LOOP FOR LAMBDA PARAMETER                                    MCS00590
      DO 500 I=1,LAM                                                   MCS00600
         DO 450 J=1,R                                                  MCS00610
            CALL GGPOS(LAMBDA(I),DPSEED,REPS(J),TRIALS,IER)            MCS00620
                                                                       MCS00630
            NSUM=0                                                     MCS00640
            DO 10 J1=1,REPS(J)                                         MCS00650
               TRIALS(J1)=TRIALS(J1)+1                                 MCS00660
10             NSUM=NSUM+TRIALS(J1)                                    MCS00670
*                                                                      MCS00680
***       LOOP FOR PROBABILITY PARAMETER                               MCS00690
            DO 400 K=1,PROB                                            MCS00700
              COUNT=COUNT+4                                            MCS00710
***         LINE COUNTER FOR PAGE BREAK                                MCS00720
              IF (COUNT.GT.55) THEN                                    MCS00730
                 WRITE (2,'(1H1)')                                     MCS00740
                 COUNT=2                                               MCS00750
                 WRITE (02,'(/T6,''LAMBDA'',T15,''REPS'',T27,''P'',    MCS00760
     C                T34,''METHOD'',T46,''ARITHMETIC'',T59,           MCS00770
     C                ''GEOMETRIC'',T71,''HARMONIC''/)')               MCS00780
              ENDIF                                                    MCS00790
*                                                                      MCS00800
               TRSUM=0                                                 MCS00810
               TRINV=0                                                 MCS00820
               TRMULT=0.                                               MCS00830
               ARCSUM=0.                                               MCS00840
               TSUM=0.                                                 MCS00850
               YSUM=0                                                  MCS00860
*                                                                      MCS00870
***       LOOP FOR REPITITION PARAMETER                                MCS00880
               DO 20 K1=1,REPS(J)                                      MCS00890
                  CALL GGBN(DPSEED,1,TRIALS(K1),P(K),YTEMP)            MCS00900
                  YSUM=YSUM+YTEMP                                      MCS00910
                  TRSUM=TRSUM+TRIALS(K1)                               MCS00920
                  TRMULT=TRMULT+ALOG(REAL(TRIALS(K1)))                 MCS00930
                  TRINV=TRINV+1./REAL(TRIALS(K1))                      MCS00940
                  Y(K1)=YTEMP                                          MCS00950
                  Z(K1)=.5*(TRIALS(K1)+.5)**.5*(ASIN(2.*Y(K1)/         MCS00960
     C                (TRIALS(K1)+1.)-1.)+ASIN(2.*(Y(K1)+1.)/         MCS00970
     C                (TRIALS(K1)+1.)-1.)+PI)                          MCS00980
                  T(K1)=Z(K1)/((TRIALS(K1)+.5)**.5)                    MCS00990
                  ARCSUM=ARCSUM+Z(K1)                                  MCS01000
20                TSUM=TSUM+T(K1)                                      MCS01010
*                                                                      MCS01020
               ZAVG=ARCSUM/REAL(REPS(J))                               MCS01030
               TAVG=TSUM/REAL(REPS(J))                                 MCS01040
***       MEAN VALUE CALCULATION                                       MCS01050
```

```
                MEAN(1)=TRSUM/REAL(REPS(J))                                    MCS01060
                MEAN(2)=EXP(TRMULT*(1./REAL(REPS(J))))                         MCS01070
                MEAN(3)=REAL(REPS(J))/TRINV                                    MCS01080
*                                                                              MCS01090
                IF (COS(TAVG).GE.0.0) THEN                                     MCS01100
                   SIGNUM=1.                                                   MCS01110
                ELSE                                                           MCS01120
                   SIGNUM=-1.                                                  MCS01130
                ENDIF                                                          MCS01140
*                                                                              MCS01150
                DO 30 K2=1,3                                                   MCS01160
                   TAVG1=ZAVG/(MEAN(K2)+.5)**.5                                MCS01170
                   IF (COS(TAVG1).GE.0.0) THEN                                 MCS01180
                      SIG1=1.                                                  MCS01190
                   ELSE                                                        MCS01200
                      SIG1=-1.                                                 MCS01210
                   ENDIF                                                       MCS01220
***        BASIC AND FTE INVERSE METHOD CALCULATION                            MCS01230
                   FTE1(K2)=.5*(1.-(SIGNUM*(1.-(ABS(SIN(TAVG)+                MCS01240
     C                (SIN(TAVG)-(1./SIN(TAVG)))/MEAN(K2)))**2.)**.5))       MCS01250
*                                                                              MCS01260
                   FTE2(K2)=.5*(1.-(SIG1*(1.-(ABS(SIN(TAVG1)+                 MCS01270
     C                (SIN(TAVG1)-(1./SIN(TAVG1)))/MEAN(K2)))**2.)**.5))     MCS01280
*                                                                              MCS01290
30                 BASIC(K2)=.5*(1.+SIN(ZAVG/((MEAN(K2)+.5)**.5)              MCS01300
     C                -(PI/2.)))                                               MCS01310
***        OUTPUT STATEMENTS                                                   MCS01320
              DO 40 K3=1,3                                                     MCS01330
                 IF (K3.EQ.1) THEN                                             MCS01340
                    WRITE (02,'(T6,F4.1,T15,I3,T25,F5.3,T35,''BASIC'',         MCS01350
     C                   T46,F7.4,T59,F7.4,T71,F7.4)')                        MCS01360
     C                   LAMBDA(I),REPS(J),P(K),(BASIC(IK),IK=1,3)            MCS01370
                 ELSE                                                          MCS01380
                    IF (K3.EQ.2) THEN                                          MCS01390
                       WRITE (02,'(T35,''FTE1'',T46,F7.4,T59,F7.4,             MCS01400
     C                      T71,F7.4)') (FTE1(IK),IK=1,3)                     MCS01410
                    ELSE                                                       MCS01420
                       WRITE (02,'(T35,''FTE2'',T46,F7.4,T59,F7.4,             MCS01430
     C                      T71,F7.4/)') (FTE2(IK),IK=1,3)                    MCS01440
                    ENDIF                                                      MCS01450
                 ENDIF                                                         MCS01460
40            CONTINUE                                                         MCS01470
*                                                                              MCS01480
400           CONTINUE                                                         MCS01490
450        CONTINUE                                                            MCS01500
500     CONTINUE                                                               MCS01510
        STOP                                                                   MCS01520
        END                                                                    MCS01530
```

LIST OF REFERENCES

1. Tucker, D.E., Loss Rate Estimation in Marine Corps Officer Manpower Models, Master's Thesis, Naval Postgraduate School, Monterey, California, September 1985.

2. Robinson, J.R., Limited Translation Shrinkage Estimation of Loss Rates in Marine Corps Manpower Models, Master's Thesis, Naval Postgraduate School, Monterey, California, March 1986.

3. Larsen, R.W., The Aggregation of Population Groups to Improve the Predictability of Marine Corps Officer Attrition Estimation, Master's Thesis, Naval Postgraduate School, Monterey, California, December 1987.

4. Carter, G.M. and Rolph, J.E., "Empirical Bayes Methods Applied to Estimating Fire Alarm Probabilities", Journal of the American Statistical Association, v. 69, no. 348, pp. 880-885, December 1974.

5. Miller, J.J., "The Inversion of the Freeman-Tukey Double Arcsine Transformation", The American Statistician, v. 32, no. 4, p. 138, November 1978.

6. Casella, G., "An Introduction to Empirical Bayes Data Analysis", The American Statistician, v. 39, no. 2, pp. 83-87, May 1985.

7. Bres, E.S. and Rowe, M.W., "Development and Analysis of Loss Rate Forecasting Techniques for the Navy's Unrestricted Line (URL) Officers", Navy Personnel Research and Development Center Technical Report 79-20, June 1979.

8. Larson, H.J., Introduction to Probability Theory and Statistical Inference, John Wiley and Sons, Inc., 1982.

9. Efron, B. and Morris C., "Stein Estimation Rule and Its Competitors--An Empirical Bayes Approach", Journal of the American Statistical Association, v. 68, pp. 117-130, 1973.

10. Fay, R.E. III and Herriot, R.A., "Estimation of Income for Small Places: An Application of James-Stein Procedures to Census Data", Journal of the American Statistical Association, V. 74, no. 366, pp. 269-277, June 1979.

11. Stein, C., "Inadmissibility of the Usual Estimator for the Mean of a Multivariate Normal Distribution", Proceedings of the Third Berkeley Symposium of Mathmatical Statistics and Probability, v. 1, Berkeley: University of California Press, pp. 197-206, 1955.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5000	1
4.	Professor Robert R. Read, Code 55Re Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5000	5
5.	Professor Paul R. Milch, Code 55Mh Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5000	1
6.	Marine Corps Representative Code 0309 Naval Postgraduate School Monterey, CA 93943-5000	1
7.	Commandant of the Marine Corps HQMC, Code MPP-30 Washington, D.C. 22134	1
8.	Commandant of the Marine Corps HQMC, Code MPI-10 Washington, D.C. 22134	1
9.	Commandant of the Marine Corps HQMC, Code MPI-20 Washington, D.C. 22134	1
10.	Commandant of the Marine Corps HQMC, Code MPI-30 Washington, D.C. 22134	1

11. Commanding Officer
Navy Personnel Research and Development Center
San Diego, CA 92152 — 2

12. Commanding Officer
ATTN: Barry Siegel
Navy Personnel Research and Development Center
San Diego, CA 92152 — 1

13. Major John R. Robinson
Center for Naval Analysis
MCOAG
4401 Ford Avenue
Alexandria, VA 22302-0268 — 1

14. Major Dewey D. Tucker
9772 High Water Court
Burke, VA 22015 — 1

15. Captain Randall W. Larsen
15820 Lazy Day Lane
Montclair Country Club
Dumgries, VA 22026 — 1

16. Major Rio M. Thalieb
J1. Mangga Kav 201
Cinere Blok A
Jakarta Selatan
INDONESIA — 1

17. Charles L. Dickinson
1924-H Treetop Lane
Birmingham, AL 35216 — 1

18. Captain Charles R. Dickinson
Staff USCINCPAC Box 15(J55)
Camp H.M. Smith, HI 96861 — 5

19. Terri Turner, Code 30
Naval Postgraduate School
Monterey, CA 93943 — 1


FILMED
8 8